

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-153061

(43)Date of publication of application : 06.06.2000

(51)Int.CI.

A63F 13/00

G06T 15/00

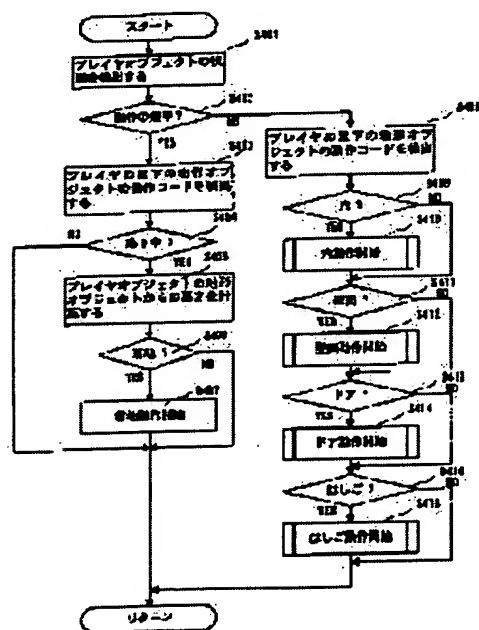
(21)Application number : 10-329805

(71)Applicant: NINTENDO CO LTD

(22)Date of filing : 19.11.1998

(72)Inventor : MIYAMOTO SHIGERU
KOIZUMI YOSHIKI
YAMADA YOICHI
IWAWAKI TOSHIO

(54) VIDEO GAME DEVICE AND INFORMATION STORAGE MEDIUM FOR VIDEO GAME



(57)Abstract:

PROBLEM TO BE SOLVED: To make simple the operation and program setting of a player by automatically executing the prescribed action of a player object according to an action code set to a topographical object.

SOLUTION: A CPU detects the action code of the topographical object near the player object in a step S408 and discriminates the kind of that topographical object in steps S409, S411, S413 and S415. When the topographical object is 'hole', the CPU executes a 'hole action' sub routine (S410). When the topographical object is 'wall', 'door' or 'ladder', similarly, the CPU executes a 'wall action', 'door action' or 'ladder

action' sub routine in a step S412, S414 or S416.

LEGAL STATUS

[Date of request for examination] 18.04.2001

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2000 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2000-153061
(P2000-153061A)

(43) 公開日 平成12年6月6日 (2000.6.6)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード (参考)
A 6 3 F 13/00		A 6 3 F 9/22	C 2 C 0 0 1
G 0 6 T 15/00		G 0 6 F 15/62	H 5 B 0 5 0
			3 6 0

審査請求 未請求 請求項の数21 O L (全 26 頁)

(21) 出願番号 特願平10-329805

(22) 出願日 平成10年11月19日 (1998. 11. 19)

(71) 出願人 000233778

任天堂株式会社

京都府京都市東山区福稲上高松町60番地

(72) 発明者 宮本 茂

京都府京都市東山区福稲上高松町60番地

任天堂株式会社内

(72) 発明者 小泉 敏晃

京都府京都市東山区福稲上高松町60番地

任天堂株式会社内

(74) 代理人 100090181

弁理士 山田 義人

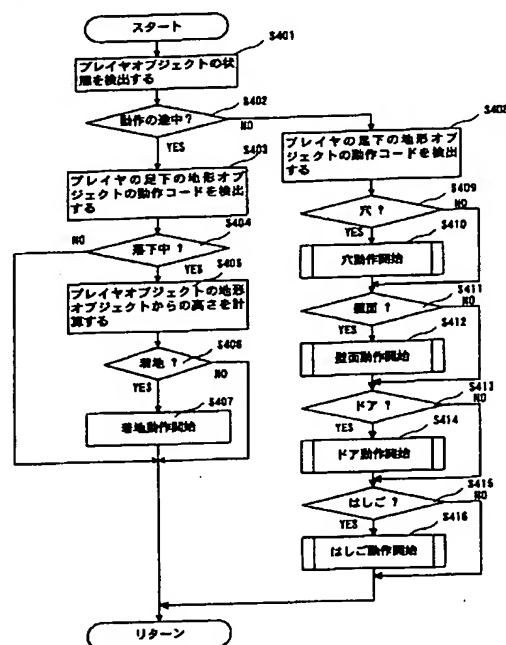
最終頁に続く

(54) 【発明の名称】 ビデオゲーム装置およびビデオゲーム用情報記憶媒体

(57) 【要約】

【構成】 CPUが、ステップS408でプレイヤオブジェクトの近傍の地形オブジェクトの動作コードを検出し、ステップS409、S411、S413、S415で、その地形オブジェクトの種類を判別する。地形オブジェクトが「穴」であった場合、CPUは「穴動作」サブルーチンを実行する (S410)。同様に、地形オブジェクトが「壁面」、「ドア」または「はしご」であった場合、CPUは、ステップS412、S414またはS416において、「壁面動作」、「ドア動作」または「はしご動作」のサブルーチンを実行する。

【効果】 地形オブジェクトに設定された動作コードに従って、プレイヤオブジェクトが自動的に所定動作を実行でき、プレイヤの操作およびプログラム設定が簡単になる。



【特許請求の範囲】

【請求項1】プレイヤオブジェクトおよび地形オブジェクトの画像データをプログラムに従って処理することによって、仮想三次元空間中の地形オブジェクト上に存在するプレイヤオブジェクトを表示するための、画像信号を発生してディスプレイに供給するビデオゲーム装置であって、

プレイヤオブジェクトの表示のためにプレイヤオブジェクト画像データを発生するプレイヤオブジェクト画像データ発生手段、および地形オブジェクトを表示するために地形オブジェクト画像データを発生する地形オブジェクト画像データ発生手段を備え、

前記地形オブジェクト画像データはプログラム制御コードを含み、

前記ビデオゲーム装置はさらに、前記プレイヤオブジェクトの位置に関連して前記プログラム制御コードを検出するプログラム制御コード検出手段、および検出したプログラム制御コードに応じて前記画像信号に変化を生じさせる画像変化手段を備える、ビデオゲーム装置。

【請求項2】前記プログラム制御コードは前記プレイヤオブジェクトの動作を制御する動作コードを含み、前記画像変化手段は、前記動作コードに応じた動作を自動的に前記プレイヤオブジェクトに行わせるアニメーションデータを出力するアニメーションデータ出力手段を含み、請求項1記載のビデオゲーム装置。

【請求項3】前記地形オブジェクトは窪みないし穴でありかつ前記動作コードが「ジャンプ」であるとき、前記アニメーションデータ出力手段は前記プレイヤオブジェクトに前記窪みないし穴をジャンプによって飛び越える動作を行わせるアニメーションデータを出力する、請求項2記載のビデオゲーム装置。

【請求項4】前記ビデオゲーム装置にはプレイヤオブジェクトの移動方向を指示する方向指示手段を含むコントローラが関連的に設けられ、前記プレイヤオブジェクトはそれによって前記移動方向へ移動し、前記ビデオゲーム装置は、さらに前記プレイヤオブジェクトの移動速度を検出する移動速度検出手段、および前記移動速度に基づいて前記プレイヤオブジェクトのジャンプ距離を計算するジャンプ距離計算手段を備え、

前記アニメーションデータ出力手段は前記プレイヤオブジェクトが前記ジャンプ距離に従ってジャンプする動作を行わせるアニメーションデータを出力する、請求項3記載のビデオゲーム装置。

【請求項5】前記地形オブジェクトが壁面でありかつ前記動作コードが「よじ登り」であるとき、前記アニメーションデータ出力手段は、前記プレイヤオブジェクトが前記壁面をよじ登る動作をするようなアニメーションデータを出力する、請求項2記載のビデオゲーム装置。

【請求項6】前記動作コードが「よじ登り」でないとき前記壁面高さを計算する壁面高さ計算手段をさらに備

え、

前記アニメーションデータ出力手段は前記プレイヤオブジェクトが前記壁面高さに応じた最適動作を行うようなアニメーションデータを出力する、請求項5記載のビデオゲーム装置。

【請求項7】前記プログラム制御コードはカメラ制御コードを含み、前記画像変化手段は前記三次元仮想空間に設けられる仮想カメラを制御するカメラ制御手段を含む、請求項1記載のビデオゲーム装置。

【請求項8】前記仮想カメラは複数の仮想カメラを含み、前記カメラ制御コードはカメラ切換コードを含み、前記カメラ制御手段は前記カメラ切換コードに応じて前記複数の仮想カメラを切り換えるカメラ切換手段を含む、請求項7記載のビデオゲーム装置。

【請求項9】前記プログラム制御コードは音声コードを含み、さらに音声データを発生する音声データ発生手段、および前記音声コードに応じて前記音声データ発生手段から出力すべき音声を制御する音声制御手段を備える、請求項1記載のビデオゲーム装置。

【請求項10】前記音声データ発生手段は複数の音声の音声データを発生でき、前記音声コードは音声切換コードを含み、前記音声制御手段は前記音声切換コードに応じて前記音声データを切り換える音声切換手段を含む、請求項9記載のビデオゲーム装置。

【請求項11】プレイヤオブジェクトおよび地形オブジェクトの画像データをプログラムに従って処理することによって、仮想三次元空間中の地形オブジェクト上に存在するプレイヤオブジェクトを表示するための、画像信号を発生し、さらに音声データをプログラムに従って処理することによって、音声信号を音声出力手段に供給するビデオゲーム装置であって、

プレイヤオブジェクトの表示のためにプレイヤオブジェクト画像データを発生するプレイヤオブジェクト画像データ発生手段、および地形オブジェクトを表示するために地形オブジェクト画像データを発生する地形オブジェクト画像データ発生手段を備え、

前記地形オブジェクト画像データはプログラム制御コードを含み、

前記ビデオゲーム装置はさらに、前記プレイヤオブジェクトの位置に関連して前記プログラム制御コードを検出するプログラム制御コード検出手段、および検出したプログラム制御コードに応じて前記音声信号に変化を生じさせる音声変化手段を備える、ビデオゲーム装置。

【請求項12】プレイヤオブジェクトおよび地形オブジェクトの画像データをプログラムに従って処理することによって、仮想三次元空間中の地形オブジェクト上に存在するプレイヤオブジェクトを表示するための、画像信号を発生してディスプレイに供給するビデオゲーム装置に適用され、かつそのビデオゲーム装置に含まれる情報処理手段によって処理されるプログラムを記憶する記憶

媒体であって、

プレイヤオブジェクトの表示のためにプレイヤオブジェクト画像データを発生するプレイヤオブジェクト画像データ発生プログラム、および地形オブジェクトを表示するために地形オブジェクト画像データを発生する地形オブジェクト画像データ発生プログラムを備え、

前記地形オブジェクト画像データはプログラム制御コードを含み、さらに前記プレイヤオブジェクトの位置に関連して前記プログラム制御コードを検出するプログラム制御コード検出プログラム、および検出したプログラム制御コードに応じて前記画像信号に変化を生じさせる画像変化プログラムを備える、ビデオゲーム装置の記憶媒体。

【請求項13】前記プログラム制御コードは前記プレイヤオブジェクトの動作を制御する動作コードを含み、前記画像変化プログラムは、前記動作コードに応じた動作を自動的に前記プレイヤオブジェクトに行わせるアニメーションデータを出力するアニメーションデータ出力プログラムを含む、請求項12記載の記憶媒体。

【請求項14】前記地形オブジェクト画像データ発生プログラムは窪みないし穴の地形オブジェクトを発生しかつ「ジャンプ」の動作コードを発生し、前記アニメーションデータ出力プログラムは前記プレイヤオブジェクトに前記窪みないし穴をジャンプによって飛び越える動作を行わせるアニメーションデータを出力する、請求項13記載の記憶媒体。

【請求項15】前記ビデオゲーム装置にはプレイヤオブジェクトの移動方向を指示する方向指示手段を含むコントローラが関連的に設けられ、前記プレイヤオブジェクトはそれによって前記移動方向へ移動し、前記記憶媒体は、さらに前記プレイヤオブジェクトの移動速度を検出する移動速度検出プログラム、および前記移動速度に基づいて前記プレイヤオブジェクトのジャンプ距離を計算するジャンプ距離計算プログラムを備え、

前記アニメーションデータ出力プログラムは前記プレイヤオブジェクトが前記ジャンプ距離に従ってジャンプする動作を行わせるアニメーションデータを出力する、請求項14記載の記憶媒体。

【請求項16】前記地形オブジェクト画像データ発生プログラムは壁面の地形オブジェクトを発生しかつ「よじ登り」の動作コードを発生し、前記アニメーションデータ出力プログラムは、前記プレイヤオブジェクトが前記壁面をよじ登る動作をするようなアニメーションデータを出力する、請求項13記載の記憶媒体。

【請求項17】前記動作コードが「よじ登り」でないとき前記壁面高さを計算する壁面高さ計算プログラムをさらに備え、

前記アニメーションデータ出力プログラムは前記プレイヤオブジェクトが前記壁面高さに応じた最適動作を行うようなアニメーションデータを出力する、請求項16記

載の記憶媒体。

【請求項18】前記地形オブジェクト画像データ発生手段はカメラ制御コードを含む地形オブジェクト画像データを発生し、前記画像変化プログラムは前記三次元仮想空間に設けられる仮想カメラを制御するカメラ制御プログラムを含む、請求項12記載の記憶媒体。

【請求項19】前記仮想カメラは複数の仮想カメラを含み、前記カメラ制御コードはカメラ切換コードを含み、前記カメラ制御プログラムは前記カメラ切換コードに応じて前記複数の仮想カメラを切り換えるカメラ切換プログラムを含む、請求項18記載の記憶媒体。

【請求項20】前記地形オブジェクト画像データ発生手段は音声コードのプログラム制御コードを含む地形オブジェクトを発生し、さらに音声データを発生する音声データ発生プログラム、および前記音声コードに応じて前記音声データ発生手段から出力すべき音声を制御する音声制御プログラムを備える、請求項12記載の記憶媒体。

【請求項21】前記音声データ発生プログラムは複数の音声の音声データを発生でき、前記音声コードは音声切換コードを含み、前記音声制御プログラムは前記音声切換コードに応じて前記音声データを切り換える音声切換プログラムを含む、請求項20記載の記憶媒体。

【発明の詳細な説明】

【0001】

【産業上の利用分野】この発明は、ビデオゲーム装置およびゲームプログラム記憶媒体に関し、特にたとえばプレイヤオブジェクトデータおよび地形オブジェクトデータによって仮想三次元空間中の地形オブジェクト上に存在するプレイヤオブジェクトを表示するための画像信号を発生してディスプレイに供給するビデオゲーム装置およびそれに用いられるゲームプログラム記憶媒体に関する。

【0002】

【従来の技術】従来のビデオゲーム機では、プレイヤがプレイヤオブジェクトをたとえばジャンプさせようとするとき、プレイヤはコントローラのジャンプボタンを押し、そのジャンプボタンの操作に応じてCPUがプレイヤオブジェクトにジャンプ動作を行わせる。つまり、プレイヤオブジェクトに窪みや穴等の障害を飛び越えさせようとするときは、プレイヤはジョイスティックや十字ボタンのような移動方向指示手段を操作しながら窪みや穴等の手前でジャンプボタンを押す。ただし、ジャンプボタンを押すタイミングつまりジャンプボタンを操作したときのプレイヤオブジェクトの位置によっては、プレイヤオブジェクトはその障害を飛び越えるのに失敗することがある。すなわち、プレイヤオブジェクトをジャンプさせて障害を越えさせるためには、ジャンプボタンの操作などに熟練を要した。

【0003】また、ジャンプ以外の動作（たとえばドア

の開閉や階段登り等)をプレイヤオブジェクトに行わせるときも、複雑なボタン操作を必要とし、プレイヤがボタン操作に気をとられるために、ゲームの進行を楽しみながらゲームをプレイすることが困難な場合もある。

【0004】

【発明が解決しようとする課題】このようなアクションゲームと呼ばれるゲームは年々難度を増しているため、プレイヤにとって難しくなり、特に初心者はその種のゲームを敬遠してしまう傾向が見えている。それゆえに、この発明の主たる目的は、新規なビデオゲーム装置およびそれに用いられるプログラム記憶媒体を提供することである。

【0005】この発明の他の目的は、プレイヤがプレイヤオブジェクトを簡単に操作できる、新規なビデオゲーム装置およびそれに用いられるゲームプログラム記憶媒体を提供することである。この発明の他の目的は、プレイヤオブジェクトが容易に障害を越える、ビデオゲーム装置およびそれに用いられるゲームプログラム記憶媒体を提供することである。

【0006】この発明の他の目的は、プレイヤオブジェクトや仮想カメラ等に自動的にジャンプ、カメラ切換等の所要動作を行わしめることのできる、ビデオゲーム装置およびそれに用いられるゲームプログラム記憶媒体を提供することである。この発明の他の目的は、簡単なプログラムによって複雑な制御ができる、ビデオゲーム装置およびそれに用いられるゲームプログラム記憶媒体を提供することである。

【0007】

【課題を解決するための手段】この発明に従ったビデオゲーム装置は、プレイヤオブジェクトおよび地形オブジェクトの画像データをプログラムに従って処理することによって、仮想三次元空間中の地形オブジェクト上に存在するプレイヤオブジェクトを表示するための、画像信号を発生してディスプレイに供給するビデオゲーム装置であって、プレイヤオブジェクトの表示のためにプレイヤオブジェクト画像データを発生するプレイヤオブジェクト画像データ発生手段、および地形オブジェクトを表示するために地形オブジェクト画像データを発生する地形オブジェクト画像データ発生手段を備え、地形オブジェクト画像データはプログラム制御コードを含み、ビデオゲーム装置はさらに、プレイヤオブジェクトの位置に関連してプログラム制御コードを検出するプログラム制御コード検出手段、および検出したプログラム制御コードに応じて画像信号に変化を生じさせる画像変化手段を備える、ビデオゲーム装置である。

【0008】プログラム制御コードはプレイヤオブジェクトの動作を制御する動作コードを含む場合、画像変化手段は、動作コードに応じた動作を自動的にプレイヤオブジェクトに行わせるアニメーションデータを出力するアニメーションデータ出力手段を含む。具体的に、地形

オブジェクトは窪みないし穴でありかつ動作コードが「ジャンプ」であるとき、アニメーションデータ出力手段はプレイヤオブジェクトに窪みないし穴をジャンプによって越える動作を行わせるアニメーションデータを出力する。

【0009】或る実施例では、ビデオゲーム装置にはプレイヤオブジェクトの移動方向を指示する方向指示手段を含むコントローラが関連的に設けられ、プレイヤオブジェクトはそれによって移動方向へ移動し、ビデオゲーム装置は、さらにプレイヤオブジェクトの移動速度を検出する移動速度検出手段、および移動速度に基づいてプレイヤオブジェクトのジャンプ距離を計算するジャンプ距離計算手段を備え、アニメーションデータ出力手段はプレイヤオブジェクトがジャンプ距離に従ってジャンプする動作を行わせるアニメーションデータを出力する。

【0010】地形オブジェクトが壁面でありかつ動作コードが「よじ登り」であるとき、アニメーションデータ出力手段は、プレイヤオブジェクトが壁面をよじ登る動作をするようなアニメーションデータを出力する。ただし、動作コードが「よじ登り」でないとき壁面高さを計算する壁面高さ計算手段をさらに備え、アニメーションデータ出力手段はプレイヤオブジェクトが壁面高さに応じた最適動作たとえば「階段登り」、「跳躍登り」、「小よじ登り」、「中よじ登り」あるいは「大よじ登り」を行うようなアニメーションデータを出力する。

【0011】この発明の実施例では、プログラム制御コードはカメラ制御コードを含み、画像変化手段は三次元仮想空間に設けられる仮想カメラを制御するカメラ制御手段を含む。なお、仮想カメラが複数の仮想カメラを含む場合、カメラ制御コードはカメラ切換コードを含み、カメラ制御手段はカメラ切換コードに応じて複数の仮想カメラを切り換えるカメラ切換手段を含む。

【0012】プログラム制御コードが音声コードを含む場合、ビデオゲーム装置はさらに音声データを発生する音声データ発生手段、および音声コードに応じて音声データ発生手段から出力すべき音声を制御する音声制御手段を備える。音声データ発生手段が複数の音声の音声データを発生できるとき、音声コードは音声切換コードを含み、音声制御手段は音声切換コードに応じて音声データを切り換える音声切換手段を含む。

【0013】なお、プログラム制御コードに応じて音声のみを制御するようにしてもよい。この場合、ビデオゲーム装置は、プレイヤオブジェクトおよび地形オブジェクトの画像データをプログラムに従って処理することによって、仮想三次元空間中の地形オブジェクト上に存在するプレイヤオブジェクトを表示するための、画像信号を発生し、さらに音声データをプログラムに従って処理することによって、音声信号を音声出力手段に供給するビデオゲーム装置であって、プレイヤオブジェクトの表示のためにプレイヤオブジェクト画像データを発生する

プレイヤオブジェクト画像データ発生手段、および地形オブジェクトを表示するために地形オブジェクト画像データを発生する地形オブジェクト画像データ発生手段を備え、地形オブジェクト画像データはプログラム制御コードを含み、ビデオゲーム装置はさらに、プレイヤオブジェクトの位置に関連してプログラム制御コードを検出するプログラム制御コード検出手段、および検出したプログラム制御コードに応じて音声信号に変化を生じさせる音声変化手段を備える、ビデオゲーム装置である。

【0014】また、ビデオゲーム装置には一般にゲームプログラムや画像データを予め記憶しておく記憶媒体が用いられる。この発明に従った記憶媒体は、プレイヤオブジェクトおよび地形オブジェクトの画像データをプログラムに従って処理することによって、仮想三次元空間中の地形オブジェクト上に存在するプレイヤオブジェクトを表示するための、画像信号を発生してディスプレイに供給するビデオゲーム装置に適用され、かつそのビデオゲーム装置に含まれる情報処理手段によって処理されるプログラムを記憶する記憶媒体であって、プレイヤオブジェクトの表示のためにプレイヤオブジェクト画像データを発生するプレイヤオブジェクト画像データ発生プログラム、および地形オブジェクトを表示するために地形オブジェクト画像データを発生する地形オブジェクト画像データ発生プログラムを備え、地形オブジェクト画像データはプログラム制御コードを含み、さらにプレイヤオブジェクトの位置に関連してプログラム制御コードを検出するプログラム制御コード検出プログラム、および検出したプログラム制御コードに応じて画像信号に変化を生じさせる画像変化プログラムを備える、記憶媒体である。

【0015】

【作用】ゲームプログラム記憶媒体に、画像データ領域が形成され、その画像データ領域には、プレイヤオブジェクトデータおよび地形オブジェクトデータが記憶される。プレイヤオブジェクトデータは、形状を示すポリゴンデータと動作状態を表すアニメーションデータとを含み、地形オブジェクトデータは形状を表すポリゴンデータおよび属性データを含む。この属性データに、プレイヤオブジェクトの動作を規定する動作コード、カメラコードあるいは音声コード等を含むプログラム制御コードが含まれる。ゲーム記憶媒体は、さらに、画像データを処理するプログラムを含み、ビデオゲーム装置はその画像データとプログラムとに従って必要に応じてコントローラからのコントローラデータを考慮してゲームを進行させ、応じて、ディスプレイ画面上には、仮想三次元空間中の地形オブジェクト上にプレイヤオブジェクトが存在するゲーム画像が表示される。

【0016】プレイヤオブジェクトが該当の地形オブジェクトに接近したまたはその地形オブジェクト上に存在するとき、地形オブジェクト画像データに含まれるプロ

ラム制御コードが検出手段によって検出される。したがって、画像変化手段は、通常のプログラムとは異なり、プレイヤオブジェクトの動作を制御し、または仮想カメラを制御する。

【0017】プレイヤオブジェクトが存在する地形オブジェクトまたはその近傍の地形オブジェクトを示す地形オブジェクトデータに動作コードが含まれる場合、その動作コードが動作コード検出手段（動作コード検出プログラム）によって検出される。そして、アニメーションデータ出力手段（アニメーションデータ出力プログラム）は、その検出した動作コードに従った動作をプレイヤオブジェクトに行わせるようなアニメーションデータを出力する。したがって、プレイヤオブジェクトは、検出した動作コードに従って自動的に最適動作を行うことができる。

【0018】具体的に、地形オブジェクトは窪みないし穴でありかつ動作コードが「ジャンプ」であるとき、アニメーションデータ出力手段（アニメーションデータ出力プログラム）はプレイヤオブジェクトに窪みないし穴をジャンプによって越える動作を行わせるアニメーションデータを出力する。プレイヤオブジェクトがコントローラの方指示手段に従って移動しているとき、移動速度検出手段（移動速度検出プログラム）がプレイヤオブジェクトの移動速度を検出し、ジャンプ距離検出手段（ジャンプ距離検出プログラム）がプレイヤオブジェクトのジャンプ距離を計算する。この場合、アニメーションデータ出力手段（アニメーションデータ出力プログラム）はプレイヤオブジェクトがジャンプ距離に従ってジャンプする動作を行わせるアニメーションデータを出力する。

【0019】地形オブジェクトが壁面でありかつ動作コードが「よじ登り」であるとき、アニメーションデータ出力手段（アニメーションデータ出力プログラム）は、プレイヤオブジェクトが壁面をよじ登る動作をするようなアニメーションデータを出力する。壁面高さ計算手段（壁面高さ検出プログラム）によって壁面高さが検出されると、その壁面高さ（H）が、 $0 < H \leq 25$ 、 $25 < H \leq 50$ 、 $50 < H \leq 100$ 、 $100 < H \leq 150$ 、または $150 < H \leq 250$ の範囲のいずれであるか判断され、アニメーションデータ出力手段（アニメーションデータ出力プログラム）は、その壁面高さ範囲に応じた最適動作を行うようなアニメーションデータを出力する。

【0020】さらに、プログラム制御コードがカメラコードである場合、たとえば仮想三次元空間中に適宜配置された複数の仮想カメラが、そのカメラコード（カメラ切換コード）によって選択的に有効化される。また、プログラム制御コードが音声コードである場合、たとえば音声データ発生手段によって発生される音声データが切り換えられる。

【0021】

【発明の効果】この発明によれば、地形オブジェクト画像データに含まれるプログラム制御コードに応じて、必要な制御、たとえばプレイヤオブジェクトの動作制御、仮想カメラの切換等の画像変化制御、あるいは音声データの切換等の音声制御を自動的に実行するようにしたので、プレイヤオブジェクトやカメラをプレイヤオブジェクトの存在する場所に応じて複雑に制御する場合であっても、そのためのプログラムの設計が簡単に行える。

【0022】たとえば、プログラム制御コードが動作コードであれば、プレイヤはプレイヤオブジェクトを非常に簡単に操作することができる。動作コードが「ジャンプ」であれば、プレイヤオブジェクトは自動的にジャンプし、したがって、プレイヤオブジェクトが容易に穴や窪みの障害を越えることができる。動作コードが「よじ登り」であれば、プレイヤオブジェクトは自動的に壁面をよじ登ることができ、動作コードが「ドア」であれば、ドアが自動的に開閉しプレイヤオブジェクトはドア内に進入できる。さらに、動作コードが「はしご」である場合、プレイヤオブジェクトは自動的にはしご登りをする。

【0023】この発明のその他の目的、特徴および利点は、添付図面に関連して行われる以下の実施例の詳細な説明から一層明らかとなろう。

【0024】

【実施例】図1に示す実施例のビデオゲームシステムは、ビデオゲーム機10と、情報記憶媒体の一例のROMカートリッジ20と、ビデオゲーム機10に接続されるディスプレイ30と、コントローラ40とを含んで構成される。コントローラ40には、コントローラバック50が着脱自在に装着される。

【0025】コントローラ40は、両手または片手で把持可能な形状のハウジング41に、複数のスイッチないしボタンを設けて構成される。具体的には、コントローラ40は、ハウジング41の左右端部および中央部に、それぞれ下方に延びて形成されるハンドル41L、41Cおよび41Rを含み、ハウジング41の上面が操作領域である。操作領域には、中央下部にアナログ入力可能なジョイスティック（以下、「アナログジョイスティック」という。）45が設けられ、左側に十字形のディジタル方向スイッチ（以下、「十字スイッチ」という。）46が設けられ、右側に複数のボタンスイッチ47A、47B、47C、47D、47Eおよび47Fが設けられる。

【0026】アナログジョイスティック45は、スティックの傾き量と方向とによって、プレイヤオブジェクト（プレイヤがコントローラ40によって操作可能なオブジェクト）の移動方向および/または移動速度ないし移動量を入力するために用いられる。十字スイッチ46は、アナログジョイスティック45に代えてプレイヤオブジェクトの移動方向を指示するために用いられる。ボ

タンスイッチ47Aおよび47Bは、プレイヤオブジェクトの動作を指示するために利用され、ボタンスイッチ47C-47Dは、三次元画像のカメラの視点を切り換えたり、プレイヤオブジェクトのスピード調節等に用いられる。

【0027】操作領域のほぼ中央部にはスタートスイッチ47Sが設けられ、このスタートスイッチ47Sは、ゲームを開始させるときに操作される。中央部のハンドル41Cの裏側にスイッチ47Zが設けられ、このスイッチ47Zは、たとえばシューティングゲームにおいてトリガスイッチとして利用される。ハウジング41の左右上部側面にはスイッチ47Lおよび47Rが設けられる。

【0028】なお、上述のボタンスイッチ47C-47Fは、カメラの視点切換え以外の用途として、シューティングまたはアクションゲームにおいてプレイヤオブジェクトの動作および/または移動速度を制御（たとえば、加速または減速）するためにも使用できる。しかしながら、これらのスイッチ47A-47F、47S、47Z、47Lおよび47Rの機能は、ゲームプログラムによって任意に定義することができる。

【0029】図2は図1実施例のビデオゲームシステムのプロック図である。ビデオゲーム機10には、中央処理ユニット（以下、「CPU」という。）11およびコプロセッサ（リアルティ・コプロセッサ；以下、「RCP」という。）12が内蔵される。RCP12には、バスの制御を行うためのバス制御回路121と、ポリゴンの座標変換や陰影処理等を行うための信号プロセッサ（リアルティ・シグナル・プロセッサ；以下、「RSP」という。）122と、ポリゴンデータを表示すべき画像にラスタライズしかつフレームメモリに記憶可能なデータ形式（ドットデータ）に変換するための描画プロセッサ（リアルティ・ディスプレイ・プロセッサ；以下、「RDP」という。）46とが含まれる。

【0030】RCP12には、外部ROM21を内蔵するROMカートリッジ20を着脱自在に装着するためのカートリッジ用コネクタ13と、ディスクドライブ29を着脱自在に装着するためのディスクドライブ用コネクタ197と、RAM14とが接続される。また、RCP12には、CPU11によって処理された音声信号および映像信号をそれぞれ出力するためのDAC（デジタル/アナログ変換器）15および16が接続される。さらに、RCP12には、1つまたは複数のコントローラ40の操作データおよび/またはコントローラバック50のデータをシリアル転送するためのコントローラ制御回路17が接続される。

【0031】RCP12に含まれるバス制御回路121は、CPU11からバスを介してパラレル信号で与えられたコマンドをパラレル/シリアル変換して、シリアル信号としてコントローラ制御回路17に供給する。ま

た、バス制御回路121は、コントローラ制御回路17から入力されたシリアル信号をパラレル信号に変換し、バスを介してCPU11へ出力する。コントローラ40から読み込まれた操作状態を示すデータ（操作信号ないし操作データ）は、CPU11によって処理されたり、RAM14に一時記憶される等の処理が行われる。換言すれば、RAM14は、CPU11によって処理されるデータを一時記憶する記憶領域を含み、バス制御回路121を介してデータの読出または書込を円滑に行うことに利用される。

【0032】音声用DAC15には、ビデオゲーム機10の後面に設けられるコネクタ19aが接続される。画像用DAC16には、ビデオゲーム機10の後面に設けられるコネクタ19bが接続される。コネクタ19aには、ディスプレイ30のスピーカ31が接続される。コネクタ19bには、テレビジョン受像機またはCRT等のディスプレイ30が接続される。

【0033】コントローラ制御回路17には、ビデオゲーム機10の前面に設けられるコントローラ用コネクタ18が接続される。コネクタ18には、接続用ジャックを介してコントローラ40が着脱自在に接続される。このように、コネクタ18にコントローラ40を接続することにより、コントローラ40がビデオゲーム機10と電氣的に接続され、相互間のデータの送受信または転送が可能とされる。

【0034】コントローラ制御回路17は、RCP12とコントローラ用コネクタ18との間でデータをシリアルで送受信するために用いられ、図3に示すように、データ転送制御回路171、送信回路172、受信回路173および送受信データを一時記憶するためのRAM174を含む。データ転送制御回路171は、データ転送時にデータフォーマットを変換するためにパラレル/シリアル変換回路とシリアル/パラレル変換回路を含み、さらにRAM174の書込/読出制御を行う。シリアル/パラレル変換回路は、RCP12から供給されるシリアルデータをパラレルデータに変換してRAM174または送信回路172に与える。パラレル/シリアル変換回路は、RAM174または受信回路173から供給されるパラレルデータをシリアルデータに変換して、RCP12に与える。送信回路172は、データ転送制御回路171から供給されるコントローラ40の信号読込のためのコマンドおよびコントローラバック50への書込データ（パラレルデータ）をシリアルデータに変換して、各コントローラ40のそれぞれに対応するチャンネルCH1~CH4へ送出する。受信回路173は、各コントローラ40に対応するチャンネルCH1~CH4から入力される各コントローラの操作データおよびコントローラバック50からの読出データをシリアルデータで受信し、パラレルデータに変換してデータ転送制御回路171に与える。データ転送制御回路171は、RCP

12から転送されたデータまたは受信回路173で受信されたコントローラデータやコントローラバック50の読出データをRAM174に書込んだり、RCP12からの命令に基づいてRAM174のデータを読み出してRCP12へ転送する。

【0035】なお、RAM174は、図示を省略しているが、各チャンネルCH1~CH4毎の記憶場所を有し、各記憶場所に当該チャンネルのコマンド、送信データおよび/または受信データがそれぞれ記憶される。図4はコントローラ40およびコントローラバック50の詳細な回路図である。コントローラ40のハウジングには、ジョイスティック45、各スイッチ46、47等の操作状態を検出しかつその検出データをコントローラ制御回路17へ転送するために、操作信号処理回路44等が内蔵される。操作信号処理回路44は、受信回路441、制御回路442、スイッチ信号検出回路443、カウンタ回路444、ジョイボート制御回路446、リセット回路447およびNORゲート448を含む。受信回路441は、コントローラ制御回路17から送信される制御信号やコントローラバック50への書込データ等のシリアル信号をパラレル信号に変換して制御回路442に与える。制御回路442は、コントローラ制御回路17から送信される制御信号がジョイスティック45のX、Y座標のリセット信号であるとき、リセット信号を発生してNORゲート448を介してカウンタ444内のX軸用カウンタ444XとY軸用カウンタ444Yの計数値をリセット(0)させる。

【0036】ジョイスティック45は、レバーの傾き方向のX軸方向とY軸方向に分解して傾き量に比例したパルス数を発生するように、X軸用とY軸用のフォトインタラプトを含み、それぞれのパルス信号をカウンタ444Xおよびカウンタ444Yに与える。カウンタ444Xは、ジョイスティック45がX軸方向に傾けられたとき、その傾き量に応じて発生されるパルス数を計数する。カウンタ444Yは、ジョイスティック45がY軸方向に傾けられたとき、その傾き量に応じて発生されるパルス数を計数する。したがって、カウンタ444Xとカウンタ444Yとの計数値によって決まるX軸とY軸の合成ベクトルによって、プレイヤオブジェクトまたは主人公キャラクタもしくはカーソルの移動方向と座標位置とが決定される。なお、カウンタ444Xおよびカウンタ444Yは、電源投入時にリセット信号発生回路447から与えられるリセット信号、またはプレイヤが所定の2つのスイッチを同時に押圧したときにスイッチ信号検出回路443から与えられるリセット信号によって、リセットされる。

【0037】スイッチ信号検出回路443は、制御回路442から一定周期（たとえばテレビジョンのフレーム周期である1/30秒間隔）で与えられるスイッチ状態を出力するためのコマンドにตอบสนองして、十字スイッチ4

6およびスイッチ47A~47Zの押圧状態によって変化する信号を読み込み、それを制御回路442へ与える。制御回路442は、コントローラ制御回路17からの操作状態データの読出指令信号に応答して、各スイッチ47A~47Zの操作状態データおよびカウンタ444Xおよび444Yの計数値を所定のデータフォーマットで送信回路445に与える。送信回路445は、制御回路442から出力されたパラレル信号をシリアル信号に変換して、変換回路43および信号線42を介してコントローラ制御回路17へ転送する。制御回路442には、アドレスバスおよびデータバスならびにポートコネクタ46を介してジョイポート制御回路446が接続される。ジョイポート制御回路446は、コントローラバック50がポートコネクタ46に接続されているとき、CPU11の命令に従ってデータの入出力（または送受信）制御を行う。

【0038】コントローラバック50は、アドレスバスおよびデータバスにRAM51を接続し、RAM51に電池52を接続して構成される。RAM51は、ゲームに関連するバックアップデータを記憶するものであり、コントローラバック50がポートコネクタ46から抜き取られても電池52からの電力供給を受けてバックアップデータを保持する。

【0039】図5はROMカートリッジ20（図1、図2）に内蔵される外部ROM21のメモリ空間を示すメモリマップである。外部ROM21は、複数の記憶領域（以下、単に「領域」と呼ぶこともある）、すなわちプログラム領域22、画像データ領域23およびサウンドメモリ領域24を含み、各種のプログラムを予め固定的に記憶している。

【0040】プログラム領域22は、ゲーム画像を処理するために必要なプログラムや、ゲーム内容に応じたゲームデータ等を記憶している。具体的には、プログラム領域22は、CPU11の動作プログラムを予め固定的に記憶するための記憶領域22a~22iを含む。メインプログラム領域22aには、後述の図7に示すゲーム等のメインルーチンの処理プログラムが記憶される。コントローラデータ判断プログラム領域22bには、コントローラ40の操作データを処理するためのプログラムが記憶される。地形オブジェクトプログラム領域22cには、プレイヤオブジェクトがその上またはその近傍に存在する地形オブジェクトの表示および制御のためのプログラムが記憶される。プレイヤオブジェクトプログラム領域22dには、プレイヤによって操作されるオブジェクト（単に、「プレイヤオブジェクト」と呼ぶ。）の表示および制御のためのプログラムが記憶される。

【0041】プログラム領域22は、さらに制御コード検出プログラム領域22eを含み、この領域22eには、地形オブジェクトの画像データに含まれる制御コード（後述）を検出するプログラムが設定されている。カ

メラ制御プログラム領域22fには、プレイヤオブジェクトを含む移動オブジェクトや背景オブジェクトを三次元空間中のどの方向および／または位置で撮影させるかを制御するためのカメラ制御プログラムが記憶される。実施例では、三次元空間中に複数の仮想カメラを設置し、したがって、カメラ制御プログラム領域22fには、第1仮想カメラから第N仮想カメラのそれぞれを個別的に制御する第1カメラ制御プログラム、第2カメラ制御プログラム、…、第Nカメラ制御プログラムが含まれる。

【0042】動作制御プログラム領域22gは、制御コード検出プログラムによって検出した制御コードに従ってプレイヤオブジェクトを動作させるように、プレイヤオブジェクトの画像データに含まれるアニメーションデータを読み出すプログラムが記憶されている。動作制御プログラムは、具体的には、各種計算プログラムを含み、計算プログラムとしては、プレイヤオブジェクトの移動速度を検出する移動速度検出プログラム、移動速度に基づいてプレイヤオブジェクトのジャンプ距離を計算するジャンプ距離計算プログラム、壁面高さを計算する壁面高さ計算プログラム等が含まれる。また、この動作制御プログラムは、動作コードないし制御コードや計算プログラムに従ってプレイヤオブジェクトの動作を決定し、その動作に応じたアニメーションデータを画像データ領域23から読み出す。したがって、動作制御プログラム22gが、画像データ領域23と協働してアニメーションデータ出力プログラムを構成する。

【0043】イメージバッファおよびZバッファ書込プログラム領域22hには、CPU11がRCP12に書込処理させるべきイメージバッファおよびZバッファへの書込プログラムが記憶される。たとえば、書込プログラム領域22hには、1つの背景画面で表示すべき複数の移動オブジェクトまたは背景オブジェクトのテクスチャデータに基づく画像データとして、色データをRAM14のフレームメモリ領域203（図6）に書き込むプログラムと、奥行データをZバッファ領域204（図6）に書き込むプログラムとが記憶される。

【0044】なお、音声処理プログラム領域22iには、効果音や音楽や音声によるメッセージを発生するためのプログラムが記憶される。画像データ領域23は、図5に示すように、2つの記憶領域23aおよび23bを含む。記憶領域23aには、プレイヤオブジェクトを表示するために、各オブジェクト毎に、複数のポリゴンの座標データおよびアニメーションデータ等の画像データをそれぞれ記憶するとともに、これらのオブジェクトを所定の位置に固定的に表示または移動表示させるための表示制御プログラムを記憶している。記憶領域23bには、地形オブジェクトを表示するために、各オブジェクト毎に、複数のポリゴンデータおよび属性データ等の画像データを記憶し、かつそれらの地形オブジェクト

の表示のための表示制御プログラムを記憶している。属性データには、プレイヤオブジェクトがすべき動作（たとえば、ジャンプ、壁登り、ドア開閉、およびはしご登り等）を示す動作コード、地形ポリゴンの種類（穴、水、砂、および溶岩等）を示す種類コード、BGMの種類を示す音楽コード、敵が存在するか否かおよび敵の種類を示す敵コード、カメラ切換を指示するカメラコード等が含まれる。これらのコードを総称して「制御コード」というが、その制御コードは、それぞれを設定する必要がある地形オブジェクトを構成するすべてのポリゴンのポリゴンデータ中に予め設定されている。なお、必要な地形オブジェクトとしては、プレイヤオブジェクトがその上に存在する地形オブジェクト、プレイヤオブジェクトが近くに存在する地形オブジェクト等が考えられる。

【0045】サウンドメモリ領域24には、場面毎に対応して、その場面に適した上記メッセージを音声で出力するためのセリフや効果音やゲーム音楽等のサウンドデータが記憶される。具体的には、後に説明するように、ゲーム音楽としてBGM1やBGM2を、そして効果音として「叫び声」等の音声データを記憶している。なお、記憶媒体ないし外部記憶装置としては、ROMカートリッジ20に代えてまたはROMカートリッジ20に加えて、CD-ROMや磁気ディスク等の各種記憶媒体を用いてもよい。その場合、CD-ROMや磁気ディスク等の光学式または磁気式等のディスク状記憶媒体からゲームのための各種データ（プログラムデータおよび画像表示のためのデータを含む）を読み出しまたは必要に応じて書き込むために、ディスクドライブ（図示せず）が設けられる。このディスクドライブは、外部ROM21と同様のプログラムデータが磁氣的または光学的に記憶された磁気ディスクまたは光ディスクに記憶されたデータを読み出し、そのデータをRAM14に転送する。

【0046】このように、プログラム領域22には、従来のビデオゲーム装置の場合と同様に、画像データ領域23に設定されている画像データを処理することによってゲーム画像信号を作成し、さらにサウンドメモリ領域24に設定されている音声データを処理することによって音声信号を作成する、プログラムが設定されている。この実施例では、さらに、画像データ領域23に記憶される画像データ、たとえば地形オブジェクトの画像データにプログラム制御コードを必要に応じて予め設定しておき、プレイヤオブジェクトの位置に応じてプログラム制御コードを検出したとき、検出したプログラム制御コードに応じて、プレイヤオブジェクトのアニメーションを変化させ、仮想カメラを切り換え、さらには音声信号を切り換える。したがって、プログラム制御コードは、プログラム制御因子ないしプログラム変更因子として機能する。

【0047】このため、プログラム制御コードを検出し

たときプレイヤオブジェクトのアニメーションを変化させたりカメラを切り換えたりすれば、通常のプログラムを実行しているときとは異なる画像変化を生じさせることができる。また、プログラム制御コードを検出したとき音声信号を切り換えるようにすれば、通常のプログラムを実行しているときとは異なる音声変化を生じさせることができる。

【0048】なお、制御コードについてより詳細に説明する。上述のように、地形オブジェクトデータは、属性データを含み、制御コードは、その属性データに含まれる。属性データは、該当の地形オブジェクトが何であるか、たとえば穴、床、壁面、階段、草原等のオブジェクトの種類を示す所定ビット数のデータである。したがって、CPU11は、その属性データを検出することによって、地形オブジェクトの種類を判断することができる。

【0049】制御コードは、属性データ中の1または2以上のビットからなり、属性データが地形オブジェクトを構成するすべてのポリゴンに含まれるので、制御データも結局すべてのポリゴンに含まれる。制御コードは上記1または2以上のビットによって、たとえば「ジャンプ」、「よじ登り」、「ドア進入」、「はしご」、「カメラ切換」、「音声切換」等の制御内容を示す。

【0050】なお、上の説明では、地形オブジェクトの種類を属性データを参照することによって判断するものとした。しかしながら、地形オブジェクトの検出方法は、次のようであってもよい。たとえば、プレイヤオブジェクトが上を移動する地形オブジェクトを床オブジェクトとして検出し、その床オブジェクトに対して90度（垂直）に設けられている地形オブジェクトを壁ないし壁面オブジェクトとして検出するようにしてもよい。この場合、プレイヤオブジェクトの上に存在する地形オブジェクトは、天井オブジェクトとして検出される。つまり、プレイヤオブジェクトとの位置関係、角度等によって地形オブジェクトの種類を判断するようにしてもよい。

【0051】いずれの場合も、プログラム制御コード（制御コード、動作コード、カメラコード、音声コード等を含む）は、属性データ中に設定される。図6はRAM14のメモリ空間全体を図解的に示したメモリマップであり、RAM14は、各種の記憶領域201～209を含む。たとえば、RAM14には、表示リスト領域201と、プログラム領域202と、1フレーム分の画像データを一時記憶するフレームメモリ（またはイメージバッファメモリ）領域203と、フレームメモリ領域のドット毎の奥行きデータを記憶するZバッファ領域204と、画像データ領域205と、サウンドメモリ領域206と、コントローラの操作状態データを記憶する領域207と、作業用（ワーキング）メモリ領域208と、レジスタ・フラグ領域209とが含まれる。各記憶領域2

01~209は、CPU11がバス制御回路121を介して、またはRCP12が直接アクセスできるメモリ空間であって、使用されるゲームによって任意の容量（又はメモリ空間）に割り当てられる。また、画像データ領域205、サウンドメモリ領域206は、ROM21の記憶領域22に記憶されている1つのゲームの全場面（又はステージ）のゲームプログラムのうち一部のデータ、たとえば或る1つのコースまたはステージに必要なゲームプログラムがプログラム領域202に転送されたとき、そのプログラムを実行するのに必要な画像データおよび音声データを一時記憶するものである。このように、或る場面に必要な各種プログラムやデータの一部分を各記憶領域202、205、206に記憶させておけば、CPU11が必要の生じる毎に直接ROM21から読み出して処理するよりも、データ処理の効率を高めることができ、画像処理速度を高速化できる。

【0052】具体的には、フレームメモリ領域203は、ディスプレイ30（図1）の画素（ピクセル又はドット）数×1画素当たりの色データのビット数に相当する記憶容量を有し、ディスプレイ30の画素に対応してドット毎の色データを記憶する。フレームメモリ領域203は、画像データ領域205に記憶されているプレイヤオブジェクト、仲間オブジェクト、敵オブジェクト、ボスオブジェクト等の移動オブジェクトと地形オブジェクトや背景（または静止）オブジェクト等の各種オブジェクトを表示する際にドット毎の色データを一時記憶する。

【0053】Zバッファ領域204は、ディスプレイ30の画素（ピクセル又はドット）数×1画素当たりの奥行きデータのビット数に相当する記憶容量を有し、ディスプレイ30の各画素に対応してドット毎の奥行きデータを記憶するものである。Zバッファ領域204は、移動及び／又は静止の各オブジェクトすなわち、画像データ領域205に記憶されているプレイヤオブジェクト、仲間オブジェクト、敵オブジェクト、ボスオブジェクト等の移動オブジェクトと地形オブジェクトや背景（または静止）オブジェクト等の各種オブジェクトを表示する際にドット毎の奥行きデータを一時記憶する。

【0054】画像データ領域205は、ROM21に記憶されているゲーム表示のための静止及び／又は移動の各オブジェクト毎に複数の集合体で構成されるポリゴンの座標データおよびテクスチャデータを記憶するものであって、画像処理動作に先立ってたとえば1コース又はステージ分のデータがROM21から転送される。なお、この画像データ領域205には、外部ROM21の画像データ領域23から必要に応じて読み出したアニメーションデータも記憶される。

【0055】サウンドメモリ領域206は、ROM21の記憶領域に記憶されている音声データ（セリフ、音楽、効果音のデータ）の一部が転送され、音声発生装置

32から発生される音声のデータとして一時記憶する。コントローラデータ（操作状態データ）記憶領域207は、コントローラ40から読み込まれた操作状態を示す操作状態データを一時記憶する。

【0056】作業用メモリ領域208は、CPU11がプログラムを実行中にパラメータ等のデータを一時記憶する。レジスタ・フラグ領域209は、レジスタ領域209rとフラグ領域209fを含む。レジスタ領域209rには図示しないが複数のレジスタが個別にデータをロードできるように形成される。レジスタ領域209rには図示しないが複数のフラグが個別にセットまたはリセットできるように形成される。

【0057】図7はこの実施例のビデオゲームシステムのメインフロー図であり、電源が投入されると、最初のステップS1において、CPU11はスタートに際してビデオゲーム機10を所定の初期状態に設定する。たとえば、CPU11は、外部ROM21のプログラム領域22に記憶されているゲームプログラムのうちの立ち上げプログラムをRAM14のプログラム領域202に転送し、各パラメータを初期値に設定した後、図7の各ステップを順次実行する。

【0058】図7のメインフロー図の動作は、たとえば1フレーム（1/60秒）毎または2ないし3フレーム毎に行われるものであり、コースをクリアするまではステップS2~S12が繰り返して実行される。コースクリアに成功することなくゲームオーバーになると、ステップS13に続いて、ステップS14においてゲームオーバー処理が行われる。コースクリアに成功するとステップS12からステップS1へ戻る。

【0059】すなわち、ステップS1において、ゲームのコース画面および／またはコース選択画面の表示が行われるが、電源投入後にゲームを開始する場合は、最初のコース画面の表示が行われる。最初のコースをクリアすると、次のコースが設定される。ステップS1に続いて、ステップS2において、コントローラ処理が行われる。この処理は、コントローラ40のジョイスティック45、十字スイッチ46、およびスイッチ47A~47Zの何れが操作されたかを検出し、その操作状態の検出データ（コントローラデータ）を読み込み、読み込んだコントローラデータをRAM14のコントローラデータ領域141に書き込む。

【0060】ステップS3において、地形オブジェクトの処理が行われる。この処理は、詳細には後に図8のサブルーチンを参照して説明するが、記憶領域22cから一部転送されたプログラムと記憶領域23（図5）から転送された地形オブジェクトのポリゴンデータとに基づいて、地形オブジェクトの表示位置およびその形状を演算する。

【0061】ステップS4では、プレイヤオブジェクトの動作を決定する処理が実行される。詳細には、図9か

ら図26を参照して後に説明するように、先に説明した制御コードないし動作コードに従ってプレイヤオブジェクトの動作を決定する。ステップS5において、プレイヤオブジェクトの表示のための処理が行われる。この処理は、基本的には、図27に示すように、プレイヤの操作するジョイスティック45の操作状態（コントローラデータ）と敵からの攻撃の有無に基づいてその姿勢、方向、形状および位置を変化させる処理である。たとえば、外部ROM21の記憶領域22e（図5）から転送されたプログラムと記憶領域23aから転送されたプレイヤオブジェクトのポリゴンデータとコントローラデータすなわちジョイスティック45の操作状態とに基づいて、変化後のポリゴンデータを演算によって求める。その結果得られた複数のポリゴンにテクスチャデータによって色を付与する。

【0062】ステップS6は、カメラ決定処理を行うステップであり、具体的には、先に説明した地形オブジェクトの出に含まれるか切換コード（制御コード）に従って、複数の仮想カメラのうちどの仮想カメラを用いて仮想三次元空間中のオブジェクトを撮影するかを決定する。後に、図27から図36を参照して詳細に説明する。ステップS7において、カメラ処理が行われる。たとえば、仮想カメラのファインディングを通して見たときの視線または視界がプレイヤがジョイスティック45によって指定したアングルとなるように、各オブジェクトに対する視点の座標を演算する。

【0063】ステップS8において、RSP122が描画処理を行う。すなわち、RCP12は、CPU11の制御の下に、RAM14の画像データ領域201に記憶されている敵オブジェクト、プレイヤオブジェクト等の移動オブジェクトや背景等の静止オブジェクトのそれぞれのテクスチャデータに基づいて、移動オブジェクトおよび静止オブジェクトの表示のための画像データの交換処理（座標変換処理およびフレームメモリ描画処理）を行う。具体的には、複数の移動オブジェクトや静止オブジェクト毎の複数のポリゴンに色を付与する。

【0064】ステップS9において、CPU11がメッセージや音楽や効果音等の音声データに基づいて、音声処理を行なう。特に、地形オブジェクトに予め設定されている音楽コード（制御コード）に従って、図37のサブルーチンに示すように、BGM等を切り換える。次のステップS10において、CPU11が、ステップS7において描画処理された結果により、RAM14のフレームメモリ領域203に記憶されている画像データを読み出す。したがって、プレイヤオブジェクト、移動オブジェクト、静止オブジェクトおよび敵オブジェクト等がディスプレイ30（図1、図2）の表示画面上に表示される。

【0065】ステップS11において、RCP12がステップS18において音声処理した結果得られる音声デ

ータを読み出すことにより、音楽および効果音または会話等の音声は出力される。ステップS12において、コースをクリアしたか否かが判断（コースクリア検出）され、コースをクリアしていなければステップS13においてゲームオーバーになったか否かが判断され、ゲームオーバーでなければステップS2へ戻り、ゲームオーバーの条件が検出されるまでステップS2～S13が繰り返される。そして、プレイヤに許容されているミス回数が所定の回数になるか、プレイヤオブジェクトのライフを所定数量使い切る等のゲームオーバー条件になったことが検出されると、続くステップS14においてゲームの継続またはバックアップデータの記憶の選択等のゲームオーバー処理が行われる。

【0066】なお、ステップS12において、コースをクリアした条件（たとえば、ボスを倒す等）が検出されると、コースクリアの処理をした後、ステップS1へ戻る。図8は、図7のステップS3で示した地形オブジェクト処理のサブルーチンであり、その最初のステップS301では、CPU11（図2）は、外部ROM21の画像データ領域23（図5）から内部RAM14の画像データ領域205（図6）に転送されている、かつそのとき必要な地形オブジェクトに対応するポリゴンデータを読み出す。このポリゴンデータ中には、先に説明したように、制御コードが必要に応じて予め設定されている。したがって、ステップS301が実行されると、その制御データも同時に読み出されることになる。なお、読み出された制御コード（動作コード、カメラ切換コード、音声コード等）を含むポリゴンデータは、内部RAM14の表示リスト領域201に一時的に保持される。

【0067】ステップS302では、内部RAM14の画像データ領域205に転送されている、地形オブジェクトに対応するテクスチャデータを読み出す。ステップS303では、地形オブジェクトに対応するカメラデータを同様に画像データ領域205から読み出す。これらのテクスチャデータおよびカメラデータも、ポリゴンデータと同様にして、表示リスト領域201に記憶される。

【0068】そして、ステップS304で地形オブジェクトを表示リスト領域201に記憶し、ステップS305において、すべての地形オブジェクトについてステップS301からステップS304の処理が実行されるかどうか判断し、もし“NO”と判定したときは、ステップS301から再び実行される。すべての地形オブジェクトの処理が終了していれば、すなわち“YES”が判断されれば、図8のサブルーチンを終了してメインルーチンにリターンする。

【0069】図7のステップS4の動作決定処理は、具体的には、図9に示すフロー図に従って実行される。すなわち、最初のステップS401では、CPU11（図2）は、プレイヤオブジェクトの状態を検出する。つま

り、プレイヤーオブジェクトが何らかの動作中であるか否かを検出する。そして、プレイヤーオブジェクトが動作の途中であれば、ステップS402において“YES”が判断され、続くステップS403に進む。

【0070】ステップS403では、CPU11は、図6に示すRAM14のレジスタ・フラグ領域209を参照して、そのプレイヤーオブジェクトの足下にある地形オブジェクトのオブジェクトデータに含まれる制御コードないし動作コードを検出する。制御コードないし動作コードは先に説明したように、図5に示す外部ROM21の地形オブジェクト領域23bに予め設定されていて、画像データ領域205に予め転送されている。その地形オブジェクトデータが各フレーム毎に表示リスト領域201に読み出される。したがって、CPU11は、その表示リスト領域201の動作コードを検出する。

【0071】続いて、CPU11は、ステップS404において、プレイヤーオブジェクトが落下中かどうか検出する。つまり、先のステップS402においてプレイヤーオブジェクトが動作中であると判断され、その動作が「落下」動作であるかどうか判断する。プレイヤーオブジェクトが落下中であれば、次のステップS405において、CPU11は、そのときのプレイヤーオブジェクトの地形オブジェクトからの高さを検出する。そして、CPU11は、ステップS406において、プレイヤーオブジェクトの地形オブジェクトからの高さが所定の高さであるとき、すなわち高さが十分低いとき、プレイヤーオブジェクトが着地すべきであると判断する。このとき、CPU11は、次のステップS407において、プレイヤーオブジェクトに着地動作を開始させる。

【0072】つまり、CPU11は、このステップS407において、外部ROM201のプレイヤーオブジェクトデータ領域23aに記憶されている着地動作のためのアニメーションデータに基づいて、プレイヤーオブジェクトの形状を変化させて、フレームメモリ領域203に色データを書込むようにRCP12を制御する。なお、このアニメーションデータは、プレイヤーオブジェクトの骨格の動きを表すデータであり、プレイヤーオブジェクトは、他のオブジェクトと同様に、そのアニメーションデータとポリゴンデータとの組合せによって表示される。したがって、同じポリゴンデータであってもアニメーションデータが変化すればプレイヤーオブジェクトの動作が変化する。そのため、このステップS407において「着地動作」のためのアニメーションデータを読み出すことによって、プレイヤーオブジェクトに着地動作させることができる。

【0073】先のステップS402においてプレイヤーオブジェクトの動作状態が「動作の途中」でないと判断したとき、CPU11は、ステップS408で、ステップS403と同様にして、プレイヤーオブジェクトの近傍（前方または足下）の地形オブジェクトの制御コードな

いし動作コードを表示リスト領域201から検出する。そして、次のステップS409では、CPU11は、プレイヤーオブジェクトの足下の地形オブジェクトの属性データを参照することによって、その地形オブジェクトが「窪み」ないし「穴」かどうか判断する。または、プレイヤーオブジェクトの進行方向に対して0度（平行もしくは水平）の床オブジェクトがあり、その床オブジェクトが下方への段差を持って形成されていることから、そのときの地形オブジェクトが窪みないし穴であると判断してもよい。

【0074】地形オブジェクトが「窪み」ないし「穴」であった場合、CPU11は、続くステップS410において、図10に示す「穴動作」サブルーチンを実行する。ステップS409において“NO”が判断されると、ステップS411において、属性コードによって、地形オブジェクトが「壁面」かどうか判断される。ただし、前述のように、プレイヤーオブジェクトの進行方向または床オブジェクトに対する角度（90度）によって壁面オブジェクトであることを検出するようにしてもよい。地形オブジェクトが「壁面」であった場合、CPU11は、続くステップS412において、図16に示す「壁面動作」サブルーチンを実行する。ステップS411において“NO”が判断されると、ステップS413において、属性コードによってもしくは床オブジェクトに対する角度によって、地形オブジェクトが「ドア」かどうか判断される。地形オブジェクトが「ドア」であった場合、CPU11は、続くステップS414において、図23に示す「ドア動作」サブルーチンを実行する。ステップS413において“NO”が判断されると、ステップS415において、属性コードによってもしくは床オブジェクトに対する角度によって、地形オブジェクトが「はしご」かどうか判断される。地形オブジェクトが「はしご」であった場合、CPU11は、続くステップS416において、図25に示す「はしご動作」サブルーチンを実行する。

【0075】ここで、図10およびそれに関連する図11から図15を参照して、「穴動作」について説明する。図10の最初のステップS417では、表示リスト領域201（図6）を参照して、その穴の手前のプレイヤーオブジェクトの足下の地形オブジェクトの動作コードないし制御コードを検出する。詳しく述べると、「穴」を構成する床オブジェクトの属性データが、1または2ビット以上の制御コードを含み、制御コードが「0」であるとき、制御コードは、デフォルトとして「ジャンプ」に設定される。また、穴を構成する床オブジェクトの制御コードとしては、他に、「ならく」、「場面切換」、「落ちない」、「進入禁止」等がある。

【0076】そして、ステップS418で検出した制御コードないし動作コードが「落ちない」コードではないとき、すなわち、制御コードないし動作コードが「ジャ

ンプ」である場合、ステップS418で“NO”が判断され、CPU11は、次のステップS419において、先のステップS405と同様の方法で、プレイヤオブジェクトのそのときの地形オブジェクトからの高さを計算によって求める。

【0077】ステップS420では、計算したプレイヤオブジェクトの高さが所定高さ、たとえば「200cm」より低いかどうか判断する。ただし、「cm」は、以下の場合も同様であるが、仮想三次元空間中においての仮想長さ単位である。このステップS420で“NO”を判断したとき、CPU11は、次のステップS421において、そのときのプレイヤオブジェクトの移動速度を計算する。そして、ステップS422では、CPU11は、ステップS419で計算した高さステップS421で計算した速さとに基づいて、プレイヤオブジェクトがジャンプすべき距離を計算する。次のステップS423では、そのジャンプ距離に従ったジャンプ動作を開始させる。

【0078】図11は穴の距離L1が短いためにプレイヤオブジェクトが穴を飛び越して向こう岸に到達するようなジャンプ動作の一例を示し、図12穴の距離L2がやや大きいためにプレイヤオブジェクトが穴を飛び越せずに穴の向こう岸に手だけをかけることができるようなジャンプ動作の一例を示し、図13は穴の距離L3が大きすぎるのでプレイヤオブジェクトが穴を飛び越せずかつ向こう岸に手さえもかけることができず、結局穴に落ちてしまうようなジャンプの一例を示す。いずれの場合にも、プレイヤオブジェクトがその上に存在する地形オブジェクトに含まれるジャンプコードに従って自動的に所要のジャンプ動作が実行される。

【0079】プレイヤオブジェクトが飛び越せる距離は、プレイヤオブジェクトの移動速さに相関する。つまり、プレイヤオブジェクトが速く走っていれば、距離Lのより大きな穴も飛び越すことができるが、プレイヤオブジェクトが歩いて移動しているときには、制御コード「ジャンプ」が設定されていても、プレイヤオブジェクトが穴を飛び越えられない場合もある。したがって、プレイヤオブジェクトが歩いている場合、プレイヤオブジェクトがジャンプできずそのまま穴の中に落ちたり、穴に落ちかけるが手だけを穴の手前側の崖にかけてぶら下がった格好になったりする。

【0080】このようなそれぞれのジャンプ動作は、先に説明したように、外部ROM221のプレイヤオブジェクトデータ領域23aから相当するアニメーションデータを読み出すことによって達成される。なお、ステップS418において“YES”が判断されたとき、すなわち、穴の手前の地形オブジェクトの制御コードないし動作コードが「落ちない」コードである場合には、CPU11は、ステップS424で、プレイヤオブジェクトが落ちない動作を開始させる。この場合、プレイヤオブ

ジェクトは、穴に落ちかけるが手だけを穴の手前側の崖にかけてぶら下がった格好になったりする。

【0081】また、ステップS420でプレイヤオブジェクトの高さが200m以下であると判断した場合には、ジャンプさせるべきではないと判断して、ステップS425において、CPU11は、プレイヤオブジェクトを落下させる動作を開始する。つまり、穴の高さないし深さが200cm（仮想長さ）以上であれば、上述のようなジャンプ動作が実行され、200cm以下であれば、図14に示すように、プレイヤオブジェクトはジャンプせずにそのまま穴内に歩を進めることになる。

【0082】ステップS409で“NO”を判断したとき、ステップS411で属性データもしくは角度を参照して地形オブジェクトの種類が「壁面」であるかどうか判断される。このステップS411において“YES”が判断されると、ステップS412において、CPU11は、プレイヤオブジェクトが壁面に直面したときの動作「壁面動作」を開始する。この壁面動作は、具体的には、図15に示すフロー図に従って実行される。

【0083】図15の最初のステップS426では、CPU11は、プレイヤオブジェクトの近傍に「壁面」である地形オブジェクトに含まれている制御コードないし動作コードが、プレイヤオブジェクトが壁面を越えて進むのを禁止する「禁止」であるかどうか判断する。「禁止」コードである場合、メインルーチンにリターンする。

【0084】壁面を構成する各ポリゴンに含まれる制御コードないし動作コードが「よじ登り」であるとき、ステップS428において、CPU11は、プレイヤオブジェクトに図16に示す壁面よじ登り動作をさせる。図16の例では、プレイヤオブジェクトが壁に接触するとプレイヤオブジェクトは壁面にくっつき、プレイヤのジョイスティック45の操作に応じて壁面上を移動する。ジョイスティック45を上側に倒せばプレイヤオブジェクトは壁面上を上になり、下に倒せば、プレイヤオブジェクトは下に移動する。そして、プレイヤオブジェクトが移動した結果制御コード「よじ登り」が設定されていない壁面位置まで達すると、プレイヤオブジェクトは、壁にくっつくことができないようになり、落下することになる。つまり、プレイヤオブジェクトと正対した壁面オブジェクトに「よじ登り」の動作コードが設定されている場合、プレイヤオブジェクトは、自動的に、壁面をよじ登る動作を実行する。ただし、プレイヤオブジェクトの移動方向は、ジョイスティック45によって決定できる。

【0085】壁面オブジェクトの制御コードないし動作コードが「禁止」でなくかつ「よじ登り」ではなくしかも壁面オブジェクトの前の床オブジェクトについてデフォルトとして制御コード「ジャンプ」が設定されている場合には、ステップS429において、CPU11は、

壁面の高さを計算する。そして、その計算した壁面高さに応じて、後述のように、プレイヤオブジェクトは自動的に最適動作を実行する。

【0086】まず、CPU11は、ステップS430において先に計算した壁面高さが0から25cmの範囲であるかどうか、すなわち、 $0 < H \leq 25$ であるかどうか判断する。この範囲の高さは壁面が非常に低いことを意味し、この場合には、プレイヤオブジェクトは階段を昇るようにしてその壁面を越えることができる。したがって、次のステップS431では、CPU11は、必要なアニメーションデータを外部ROM21すなわちRAM14から読み出して、プレイヤオブジェクトに図17に示す「階段登り」動作を開始させる。図17の例では、越えるべき壁面の高さが小さいので、プレイヤオブジェクトは、床オブジェクトに設定されている制御コード「ジャンプ」に従って1歩1歩が段を踏むような動作で、壁面である階段を越えることができる。この場合、各壁面オブジェクトすなわち階段の手前の床オブジェクトに、図17に示すように、制御コード「ジャンプ」が予め設定されている。

【0087】CPU11は、続くステップS432において壁面高さが25cmから50cmの範囲であるかどうか、すなわち、 $25 < H \leq 50$ であるかどうか判断する。この範囲の高さは壁面が低いことを意味し、この場合には、プレイヤオブジェクトはジャンプによってその壁面を越えることができる。したがって、次のステップS433では、CPU11は、必要なアニメーションデータをROM21すなわちRAM14から読み出して、プレイヤオブジェクトに図18に示す「ジャンプ」動作を開始させる。図18の例では、プレイヤオブジェクトは越えるべき壁面の手前でジャンプし、そのまま壁面の上に着地することによって壁面を越える。この場合も、壁面オブジェクトの手前の地形オブジェクト、すなわち床オブジェクトに、図18に示すように、制御コード「ジャンプ」が予め設定されている。

【0088】ステップS434では、CPU11は、壁面高さが50cmから100cmの範囲であるかどうか、すなわち、 $50 < H \leq 100$ であるかどうか判断する。この範囲の高さは壁面がやや高いことを意味し、この場合には、プレイヤオブジェクトは小よじ登りによってその壁面を越えることができる。したがって、次のステップS435で、CPU11は、必要なアニメーションデータを読み出して、プレイヤオブジェクトに図19に示す「小よじ登り」動作を開始させる。図19の「小よじ登り」の例では、プレイヤオブジェクトは対象である壁面の上端に手を掛け、その手による懸垂力と足によるジャンプ力とで、その壁面の上端に体を押し上げる。この場合、壁面の手前の床に、図19に示すように、制御コード「ジャンプ」が予め設定されている。

【0089】ステップS436では、CPU11は、壁

面高さが100cmから150cmの範囲であるかどうか、すなわち、 $100 < H \leq 150$ であるかどうか判断する。この範囲の高さは壁面が高いことを意味し、この場合には、プレイヤオブジェクトは中よじ登りによってその壁面を越えることができる。したがって、次のステップS437で、CPU11は、必要なアニメーションデータを読み出して、プレイヤオブジェクトに図20に示す「中よじ登り」動作を開始させる。図20の「中よじ登り」の例では、壁面の手前の床オブジェクトに含まれる「ジャンプ」コードにตอบสนองして、プレイヤオブジェクトは対象である壁面の手前で小さくジャンプすることによって壁面の上端に手を掛け、そのときプレイヤオブジェクトの足は浮いた状態であるので、その手による懸垂力によってのみ、その壁面の上端に体を引き上げる。

【0090】そして、ステップS438では、CPU11は、壁面高さが150cmから250cmの範囲であるかどうか、すなわち、 $150 < H \leq 250$ であるかどうか判断する。この範囲の高さは壁面が非常に高いことを意味し、この場合には、プレイヤオブジェクトは大よじ登りによってその壁面を越えることができる。したがって、次のステップS439で、CPU11は、プレイヤオブジェクトに図21に示す「大よじ登り」動作を開始させる。図21の「大よじ登り」の例では、プレイヤオブジェクトは対象である壁面の手前の床オブジェクトの制御コード「ジャンプ」に応じて大きくジャンプしてその壁面の上端に手を掛け、そのときプレイヤオブジェクトの足は浮いた状態であるので、その手による懸垂力によってのみ、その壁面の上端に体を引き上げる。

【0091】このようにして、プレイヤオブジェクトの存在する地形オブジェクトまたはその近傍の地形オブジェクトのオブジェクトデータに含まれる制御コードないし動作コードを検出して、CPU11はプレイヤオブジェクトに対して、制御コードないし動作コードに応じた動作、実施例であれば、壁面越え動作をさせる。なお、壁面オブジェクトに含まれる制御コードないし動作コードが「よじ登り」であるときは、先に説明した「ジャンプ」ではなく、「よじ登り」によって壁面を越える。また、壁面オブジェクトに「禁止」コードが埋設されているならば、プレイヤオブジェクトは、その壁面を越えることができない。

【0092】図22に、図9のステップS414に示す「ドア動作」のためのサブルーチンが示される。この図22の最初のステップS440では、CPU11は、RAM14のコントローラデータ領域207を参照して、Aボタン47A（図1）がプレイヤによって送信されるかどうか判断する。Aボタン47Aが操作されていないならば、プレイヤオブジェクトをドア内に進入させる意思がプレイヤにはないと判断し、そのままメインルーチンにリターンする。

【0093】そして、ステップS440においてもし

“YES”が判断されると、CPU11は、次のステップS441において、画像データ領域205（図6）を参照して、ドアを構成するポリゴンの座標位置から、ドアの位置を検出する。次のステップS442では、ステップS441で検出したドアの位置に基づいて、プレイヤオブジェクトがドアを開く位置に正しく位置決めされるように、プレイヤオブジェクトの位置を補正する。変更されたプレイヤオブジェクトの位置は、画像データ領域205に記録される。その後、ステップS443において、ドア動作を実行する。すなわち、CPU11は、画像データ領域205から必要なポリゴンデータやアニメーションデータを読み出して、プレイヤオブジェクトが、ドアノブを持ち、ドアを開き、ドア内に入り、そしてドアを閉じる、という一連のドア動作を実行させる。図23は、このドア動作において、プレイヤオブジェクトがドア内に進入しようとする状態を図解的に示す。つまり、図9のステップS413で実行される「ドア動作」では、図23に図示した地形オブジェクト内に予め設定されるれている「ドアコード」に従って、プレイヤオブジェクトにドアの開閉動作を自動的に行わせる。

【0094】なお、上述の実施例では、ドアの直前の床オブジェクトに制御コードないし動作コードとして「ドア」が設定されるれているものとして説明した。これに対して、ドアオブジェクトに「ドア」コードを設定するようにしてもよい。図9のステップS416で実行される「はしご動作」の詳細が図24に示される。図24の最初のステップS444では、画像データ領域205（図6）を参照して、はしごを構成するポリゴンの座標位置から、はしごの位置を検出する。次のステップS445では、ステップS444で検出したはしごの位置に基づいて、プレイヤオブジェクトがはしごの下に位置決めされるように、プレイヤオブジェクトの位置を補正する。変更されたプレイヤオブジェクトの位置は、画像データ領域205に記録される。その後、ステップS446において、はしご動作を実行する。すなわち、CPU11は、画像データ領域205から必要なポリゴンデータやアニメーションデータを読み出して、プレイヤオブジェクトが、はしごに手および足を掛け、はしご上において交互に左右の手および足を運ぶ、という一連のはしご登り動作を実行させる。図25は、このはしご動作において、プレイヤオブジェクトがはしごの途中まで上った状態を図解的に示す。つまり、図9のステップS416で実行される「はしご動作」では、図25に図示した地形オブジェクトすなわち壁面オブジェクト内に予め設定されるれている「はしご」コードに従って、プレイヤオブジェクトにはしご登り動作を自動的に行わせる。

【0095】より詳しく説明すると、図25の例では、はしごを構成する壁面オブジェクトの手前の床オブジェクトに制御コード「進入禁止」が設定されていないので、プレイヤオブジェクトは、壁面オブジェクトの制御

コード「はしご」に従って「はしご登り」動作をすることができ。つまり、床オブジェクト内の制御コードが「進入禁止」でない限り、はしご登り動作に影響しない。そして、「はしご」コードが壁面に設定されていると、プレイヤオブジェクトは、はしごの壁面に正対したとき、自動的に、はしごをつかむ。その後プレイヤがジョイスティック45（図1）を上へ倒すとプレイヤオブジェクトははしご上へ上へ、下へ倒すと、下に、それぞれ移動する。そして、プレイヤオブジェクトがはしごの最上位置に達すると、プレイヤオブジェクトは、自動的にその部分の床に降り立つ。また、プレイヤオブジェクトが上の床からはしごの場所に進んできた場合には、はしごの裏の壁面オブジェクトに設定されている「はしご進入」コードを検出し、そのコードに従ってプレイヤオブジェクトは、はしごを降りる。

【0096】このようにして、この実施例によれば、プレイヤオブジェクトが存在する地形オブジェクト中に予め含まれておいた制御コードないし動作コードに応じて、自動的に、プレイヤオブジェクトをして異なる動作をさせることができる。したがって、プレイヤオブジェクトの動作制御のためのプログラムの設定が極めて簡単に行える。

【0097】なお、図26に示すフロー図は図7のメインルーチンのステップS5のプレイヤオブジェクト処理動作を示し、最初のステップS501で、CPU11は、プレイヤオブジェクトが動作途中であるかどうか判断し、もし、動作途中であれば、プレイヤオブジェクトがその動作を継続するように、プレイヤオブジェクトの位置およびポーズを決定する。ポーズは先に説明したアニメーションデータで決まる。

【0098】プレイヤオブジェクトが動作途中でないときは、CPU11は、続くステップS503において、コントローラ40に含まれるジョイスティック45（図1、図4）の操作状態を検出する。続いて、ステップS503、S504およびS505において、ジョイスティック45の操作状態に応じて、それぞれ、プレイヤオブジェクトの移動方向、移動速度および位置およびポーズを決定する。そして、ステップS502を経た後と同様に、ステップS507において、プレイヤオブジェクトをRAM14の表示リスト領域201（図6）に登録する。応じて、プレイヤオブジェクトがジョイスティック45の操作状態に応じて表示されることになる。

【0099】図9のメインルーチンのステップS6でのカメラ決定処理を図27および関連の各図を参照して詳細に説明する。図27の最初のステップS601では、CPU11は、画像データ領域205のデータを参照し、プレイヤオブジェクトの位置に基づいて、そのプレイヤオブジェクトが存在する直下の地形オブジェクトのオブジェクトデータに予め設定されるれている制御コード（カメラコード）を検出する。そして、ステップS6

02, S604, S606, S608, または S610 のそれぞれにおいて、その検出した制御コードが、第1カメラコード、第2カメラコード、第3カメラコード、第4カメラコード、または第5カメラコードのいずれであるか判断する。

【0100】ここで、図28に基づいて、実施例での仮想三次元空間中に配置された第1カメラ、第2カメラ、第3カメラ、第4カメラおよび第5カメラについて説明する。図28の例では、平面矩形の空間のほぼ中央に縦長に壁が設けられ、その壁の一部にドアが形成されている。そして、ドアの一方側（ドアが開く方向の側）にドアに向けられた第3カメラが固定的に設置されている。ドアの反対側には、第4カメラが設置されていて、その第4カメラは、ドアを開けて進入してくるブレイヤオブジェクトを撮影するズームカメラとして設定されている。さらに、空間の2つのコーナーに第2カメラおよび第5カメラがそれぞれ個別に固定的に設定されている。第1カメラは、ブレイヤオブジェクトに追従して移動する移動カメラとして設定されている。このようにして5つの仮想カメラが三次元空間中に設定されているこの実施例を前提として、以下のカメラ制御を説明する。ただし、具体的なカメラの数、カメラの配置あるいは各カメラの機能ないし役割（固定、移動、またはズーム等）は、必要に応じて適宜変更され得ることはいうまでもない。なお、図28において、各ブロック（矩形の枠目）のそれぞれに記入されている「第1カメラ」、「第2カメラ」、…、「第5カメラ」の文字は、この三次元空間中の地形オブジェクトに予め設定されている制御コードすなわちカメラコードを示す。したがって、ブレイヤオブジェクトがそのブロックに存在するとき、ブレイヤオブジェクトは、そのブロックに設定されているカメラコードに相当するカメラによって撮影される。

【0101】図27に戻って、ステップS602で第1カメラコードで検出されると、続くステップS603では、第1カメラ制御プログラムが選択的に設定される。カメラ制御プログラムは、先に説明したように、外部ROM21のカメラ制御プログラム領域22f（図5）に設定されていて、必要に応じて、内部RAM14のプログラム領域202に転送されている。したがって、CPU11は、ステップS603では、RAM14のプログラム領域202（図6）から第1カメラ制御プログラムを読み出す。

【0102】第1カメラ制御プログラムは、第1カメラのための制御プログラムであり、その第1カメラは、先に述べたように、ブレイヤオブジェクトに追従して移動する。そこで、図29に詳細に示す第1カメラ制御プログラムでは、まず、ステップS612において、画像データ領域205（図6）のデータを参照して、ブレイヤオブジェクトの位置を検出する。次のステップS613では、CPU11は、第1カメラの位置を、ブレイヤオ

ブジェクトから第1カメラのでの距離が一定になるように、決定する。ついで、ステップS614では、第1カメラの撮影方向をブレイヤオブジェクトの方向に向ける。したがって、第1カメラは、図30に示すように、ブレイヤオブジェクトの後姿を一定距離から撮影することになる。

【0103】ステップS605（図27）で実行される第2カメラ制御プログラムでは、図31に示すように、最初のステップS615において、先のステップS612（図29）と同様にして、ブレイヤオブジェクトの位置を検出し、ついで、ステップS616において、第2カメラの撮影方向をブレイヤオブジェクトに向ける。つまり、第2カメラでは、図28に示す固定位置から、ブレイヤオブジェクトを撮影することになる。

【0104】なお、第5カメラも第2カメラと同様に、固定カメラであるため、ステップS611で選択される第5カメラ制御プログラムは、図31の第2カメラ制御プログラムと同じである。第3カメラは、図28に示したように、ドアの手前側に固定的に設けられている。したがって、第3カメラは、ドアを出入りするブレイヤオブジェクトを一定距離から撮影するだけである。そのため、ステップS607（図27）の第3カメラ制御プログラムは、図32のステップS617を含み、このステップS617では、第3カメラの撮影方向をドアの方向に設定する。したがって、ブレイヤオブジェクトがそのドアを出入りする様子が、図33に示すように、第3カメラによって撮影される。

【0105】図27のステップS609で実行される第4カメラ制御プログラムが図34に詳細に示される。第4カメラが選択されるのは、図28からよくわかるように、ブレイヤオブジェクトがドアを通して進入したブロックに設定されている第4カメラコードを検出したときである。そして、図34の最初のステップS618では、その第4カメラコードを検出してステップS609に入ってから、すなわちカメラ切り換えが行われてからのフレーム数を検出する。これは、第4カメラによってブレイヤオブジェクトを撮影する方法が2通りあるからである。そして、フレーム数が所定数以下であるとき、すなわちカメラ切換の直後である場合、ステップS619において“YES”が判断される。この場合、CPU11は、ステップS620において、予め設定された所定の位置から、ドア内に進入したブレイヤオブジェクトを第4カメラで撮影するように、第4カメラを制御する。ステップS620で第4カメラによって撮影されたブレイヤオブジェクトが図35に図解される。図35からわかるように、図28に図示する位置に固定的に設定された第4カメラは、ステップS620では、すなわちカメラ切り換え直後では、ドア内に進入したブレイヤオブジェクトを遠景中に撮影する。つまり、ブレイヤオブジェクトを含む比較的広い範囲を第4カメラが撮影す

る。したがって、この実施例のように、ドア内にプレイヤオブジェクトが進入したときには、全景を表示することによって、主人公たるプレイヤオブジェクトが現在どのような場所に存在することになったのかが、プレイヤに容易に理解できる。

【0106】カメラ切換直後ではないがカメラ切換から所定フレーム数すなわち所定時間経過していないときには、ステップS621で“NO”が判断される。この場合、続くステップS622において、CPU11は、図36に示すように、プレイヤオブジェクトを近景のなかで撮影するように、第4カメラをズームアップする。つまり、プレイヤオブジェクトを含む比較的狭い範囲を撮影する。

【0107】所定フレーム数が経過していれば、ステップS621において“YES”が判断され、この場合、CPU11は、ステップS623に示すように、第4カメラから第1カメラに切り換える。このようにして、この実施例によれば、プレイヤオブジェクトが存在する地形オブジェクト中に予め含ませておいた制御コードないしカメラコードに応じて、自動的に、プレイヤオブジェクトを撮影するカメラおよびその機能を切り換えることができる。したがって、煩雑なカメラ切換を必要とする場合であっても、そのためのプログラムの設定が極めて簡単に行える。また、プレイヤオブジェクトの位置(X-Y座標位置)に応じてカメラを切り換える場合であれば、そのX-Y座標位置が同じであれば、Z座標すなわち高さのいかに拘わらず同一のカメラ切換が行われる。これに対し、この実施例の方法では、地形オブジェクトにカメラ切換コードを埋設しているので、同じX-Y平面内ではあるが高さ(Z)が異なる場合には、異なる地形オブジェクトすなわちカメラコードを設定することができるため、異なるカメラを用いることができる。つまり、実施例によれば、三次元的なカメラ切換が可能になる。

【0108】なお、いずれのステップS620、S622およびS623が終了した後は、プロセスは、メインルーチンにリターンする。図37を参照して、図7のメインルーチンにおけるステップS9すなわち音声処理について詳細に説明する。この音声処理においても、先に説明した制御コードを利用する。したがって、図37の最初のステップS624では、画像データ領域205を参照して、プレイヤオブジェクトがその上に存在する地形オブジェクトに設定されている制御コードすなわち音楽コードを検出する。そして、ステップS625において、その制御コードすなわち音楽コードがBGM1であるかどうか判断する。音楽コードBGM1は、第1BGM(Back-Ground Music)を選択するためのコードであり、したがって、ステップS625で“YES”が判断された後には、ステップS626において、CPU11は、図6に示すサウンドメモリ領域206から第1BGM

Mのための音楽ないし音声データを読み出し、それをバス制御回路121(図2)に出力する。

【0109】ステップS625で“NO”が判断されると、ステップS627では、制御コードすなわち音楽コードがBGM2であるかどうか判断する。音楽コードBGM2は、第2BGMを選択するためのコードであり、したがって、ステップS627で“YES”が判断された後には、ステップS628において、CPU11は、サウンドメモリ領域206から第2BGMのための音楽ないし音声データを読み出し、それをバス制御回路121に出力する。

【0110】ステップS625およびS627でともに“NO”が判断された後は、ステップS629において、CPU11は、制御コードすなわち音楽コードが「叫び声」であるかどうか判断する。音楽コード「叫び声」は、叫び声の効果音を発生させるためのコードであり、したがって、ステップS629で“YES”が判断された後には、ステップS630において、CPU11は、サウンドメモリ領域206から「叫び声」のための音声データを読み出し、それをバス制御回路121に出力する。

【0111】なお、制御コードないし音楽コードもしくは音声コードが先に挙げたいずれのものとも異なるときは、ステップS631において、その他の音声データが設定される。このようにして、この実施例によれば、プレイヤオブジェクトが存在する地形オブジェクト中に予め含ませておいた制御コードないし音声コードに応じて、自動的に、そのとき発生する音声データを切り換えることができる。したがって、煩雑な音声切り換え制御を必要とする場合であっても、そのためのプログラムの設定が極めて簡単に行える。

【図面の簡単な説明】

【図1】この発明の一実施例のビデオゲームシステムを示す概略図解図である。

【図2】図1システムのビデオゲーム機を詳細に示すブロック図である。

【図3】図2ビデオゲーム機のコントローラ制御回路をより詳細に示すブロック図である。

【図4】図2ビデオゲーム機のコントローラおよびコントローラバックを詳細に示すブロック図である。

【図5】図2ビデオゲーム機の外部ROMのメモリマップを示す図解図である。

【図6】図2ビデオゲーム機のRAMのメモリマップを示す図解図である。

【図7】図1実施例の全体の動作を示すフロー図である。

【図8】図7フロー図の地形オブジェクト処理を詳細に示すフロー図である。

【図9】図7フロー図の動作決定処理の一部を詳細に示すフロー図である。

【図10】図9フロー図の穴の場合の動作決定処理を詳細に示すフロー図である。

【図11】図10フロー図で達成されるジャンプ（大ジャンプ）動作の一例を示す図解図である。

【図12】図10フロー図で達成されるジャンプ（中ジャンプ）動作の一例を示す図解図である。

【図13】図10フロー図で達成されるジャンプ（小ジャンプ）動作の一例を示す図解図である。

【図14】図10フロー図の「落ちない」動作の一例を示す図解図である。

【図15】図9フロー図の壁面の場合の動作決定処理を詳細に示すフロー図である。

【図16】図15フロー図で達成される壁よじ登り動作の一例を示す図解図である。

【図17】図15フロー図で達成される階段登り動作の一例を示す図解図である。

【図18】図15フロー図で達成される跳躍登り動作の一例を示す図解図である。

【図19】図15フロー図で達成される小よじ登り動作の一例を示す図解図である。

【図20】図15フロー図で達成される中よじ登り動作の一例を示す図解図である。

【図21】図15フロー図で達成される大よじ登り動作の一例を示す図解図である。

【図22】図9フロー図のドア動作を詳細に示すフロー図である。

【図23】図22フロー図で達成されるドア動作の一例を示す図解図である。

【図24】図9フロー図のはしご動作を詳細に示すフロー図である。

【図25】図24フロー図で達成されるはしご動作の一例を示す図解図である。

【図26】図7フロー図のプレイヤーオブジェクト処理を

詳細に示すフロー図である。

【図27】図7フロー図のカメラ決定処理を詳細に示す図解図である。

【図28】図27フロー図のカメラ決定処理の前提となるカメラ配置の一例を示す図解図である。

【図29】図27フロー図の第1カメラ制御プログラムを詳細に示すフロー図である。

【図30】図29フロー図に従って第1カメラで撮影したプレイヤーオブジェクトを示す図解図である。

【図31】図27フロー図の第2カメラ（第5カメラ）制御プログラムを詳細に示すフロー図である。

【図32】図27フロー図の第3カメラ制御プログラムを詳細に示すフロー図である。

【図33】図32フロー図に従って第3カメラで撮影したプレイヤーオブジェクトを示す図解図である。

【図34】図27フロー図の第4カメラ制御プログラムを詳細に示すフロー図である。

【図35】図34フロー図に従って第4カメラで撮影したプレイヤーオブジェクトを示す図解図である。

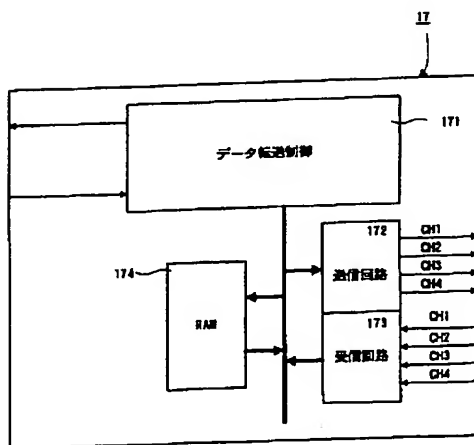
【図36】図32フロー図に従って第4カメラで撮影したプレイヤーオブジェクトを示す図解図である。

【図37】図7フロー図の音声処理を詳細に示すフロー図である。

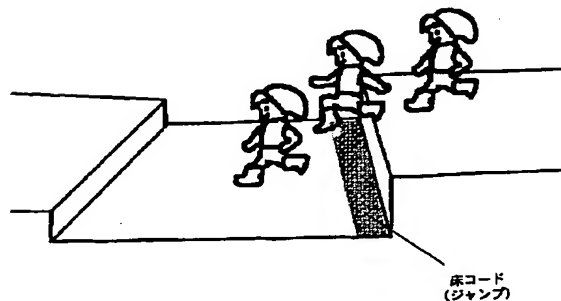
【符号の説明】

- 10 …ビデオゲーム機
- 11 …CPU
- 12 …リアルティコプロセサ
- 14 …RAM
- 21 …外部ROM
- 22 …プログラム領域
- 23 …画像データ領域
- 40 …コントローラ
- 45 …ジョイスティック

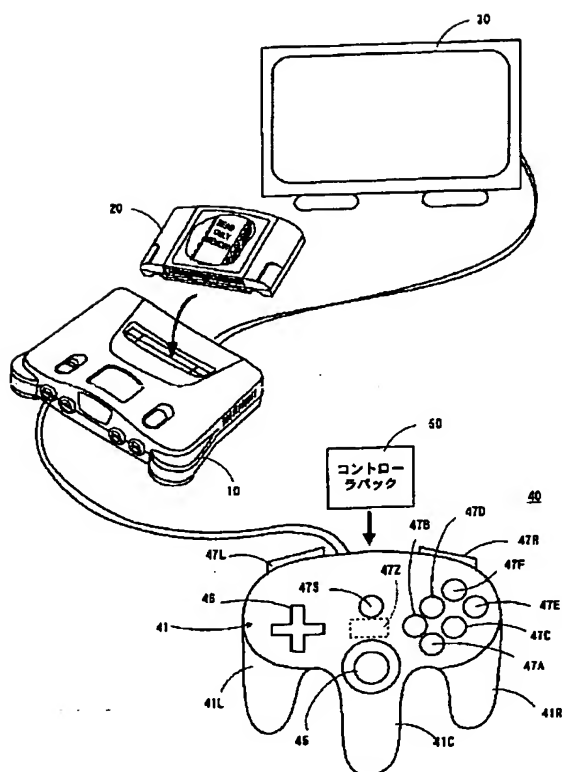
【図3】



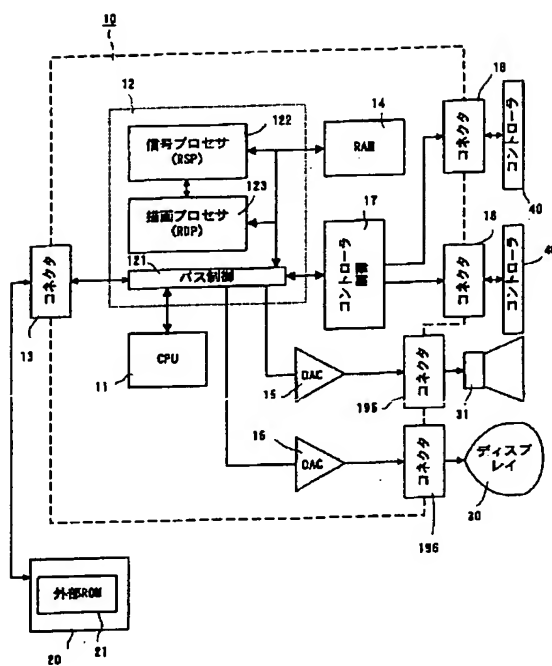
【図14】



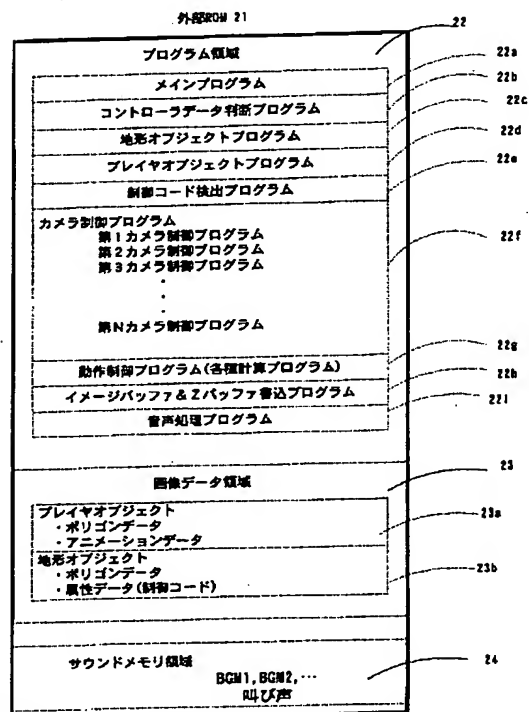
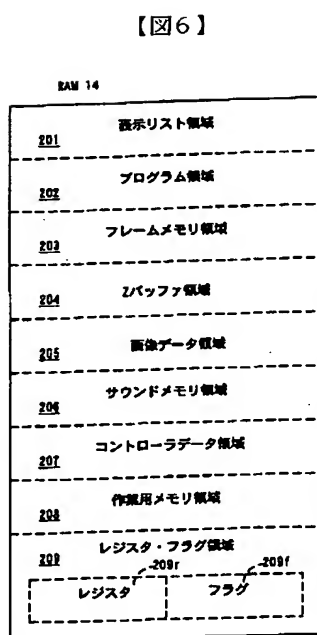
【図1】



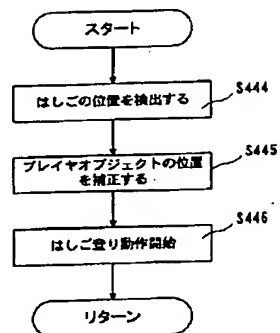
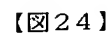
【図2】



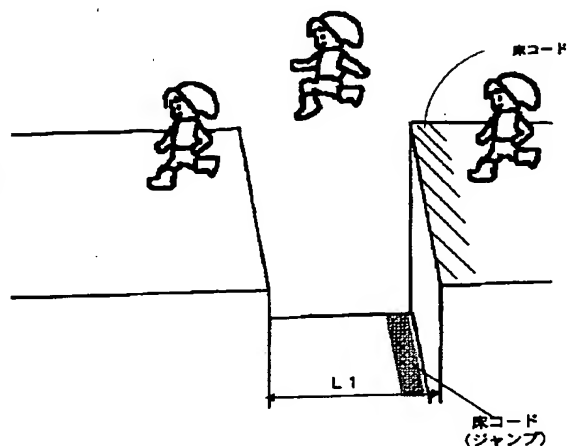
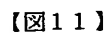
【図5】



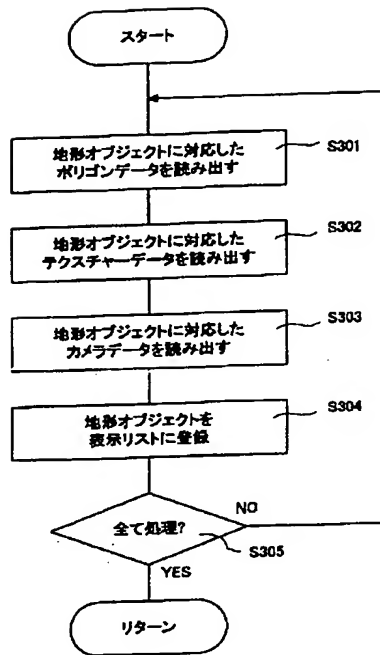
【図7】



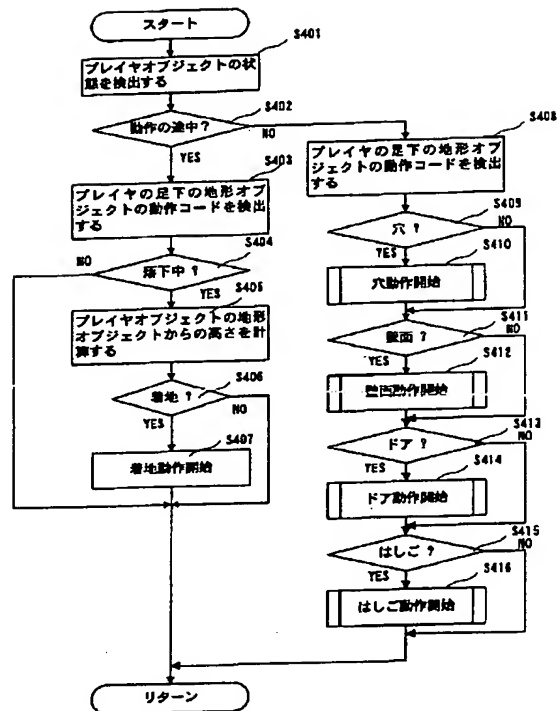
【図10】



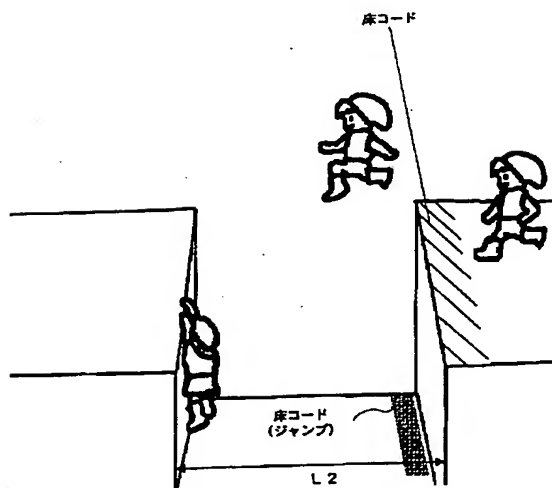
【図8】



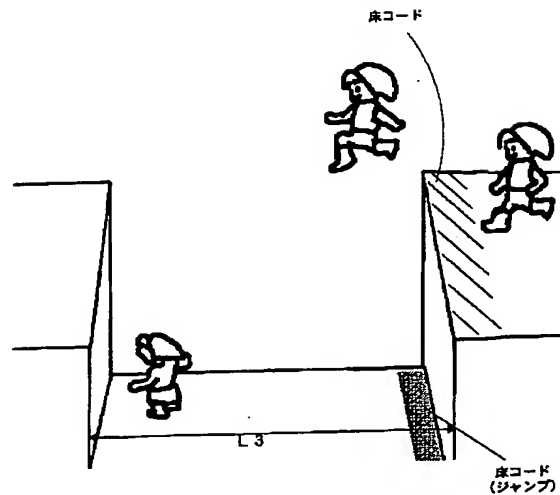
【図9】



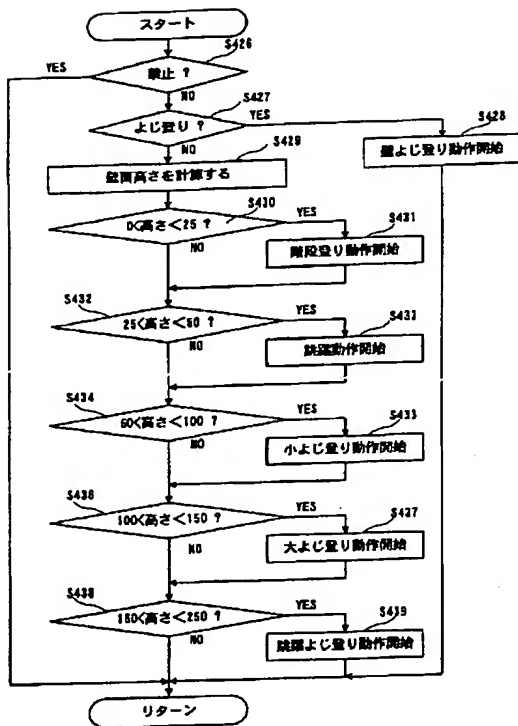
【図12】



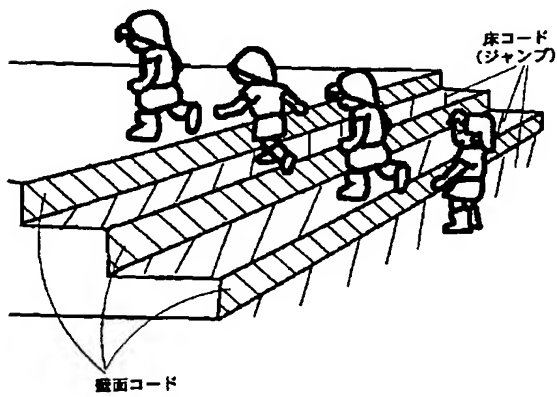
【図13】



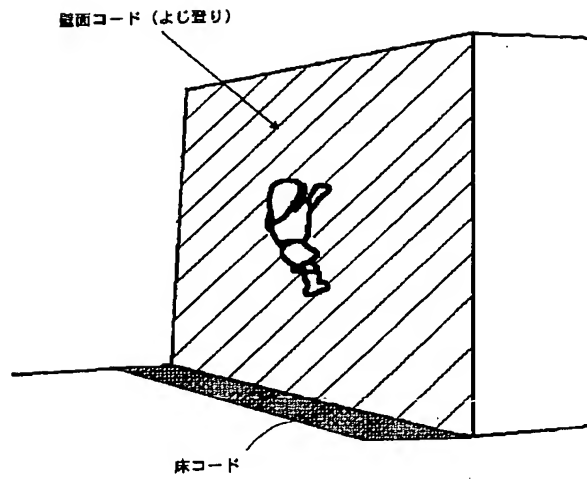
【図15】



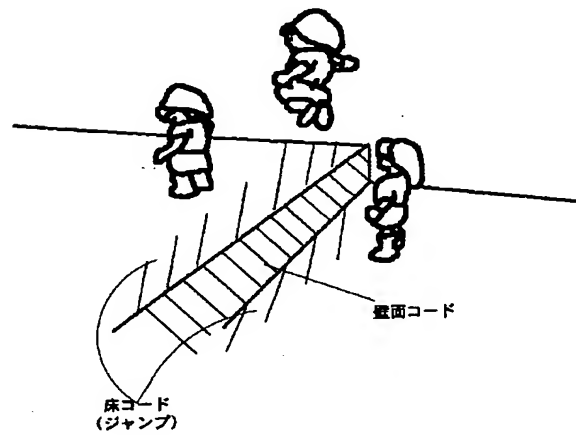
【図17】



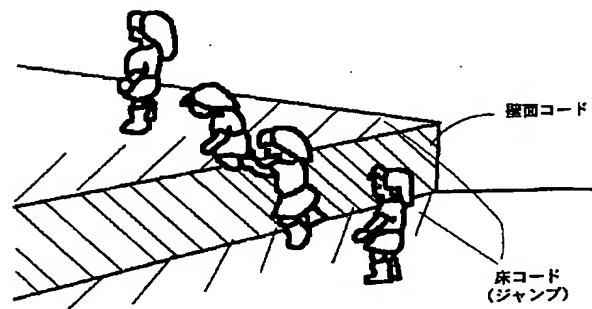
【図16】



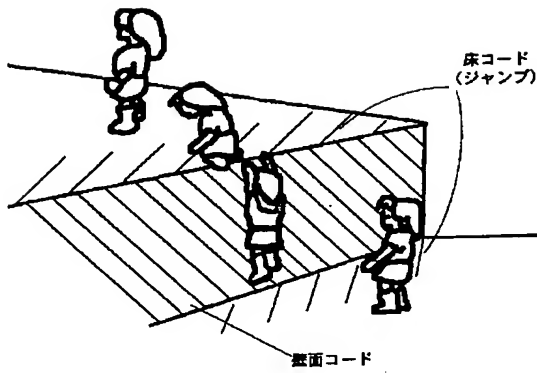
【図18】



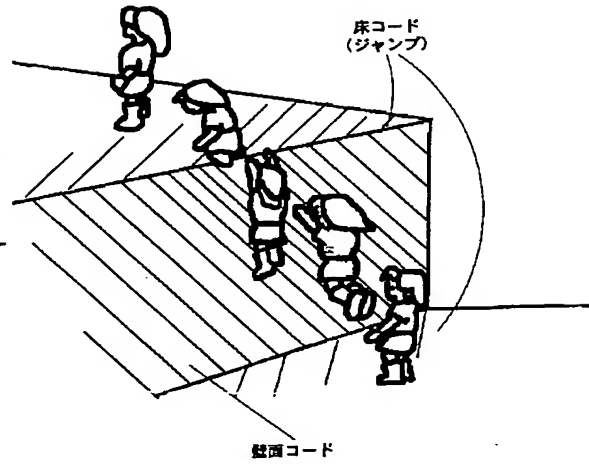
【図19】



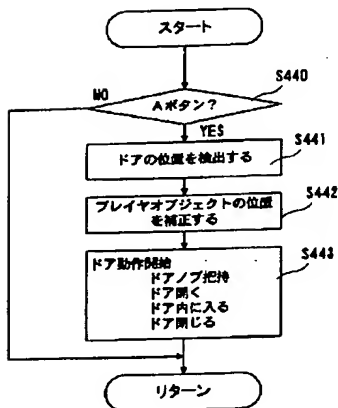
【図20】



【図21】



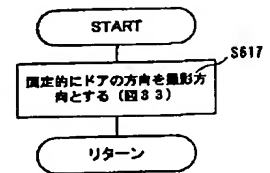
【図22】



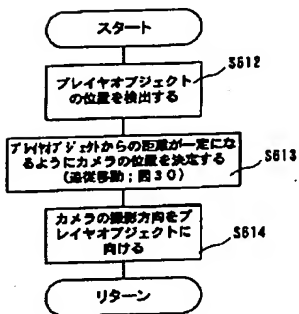
【図23】



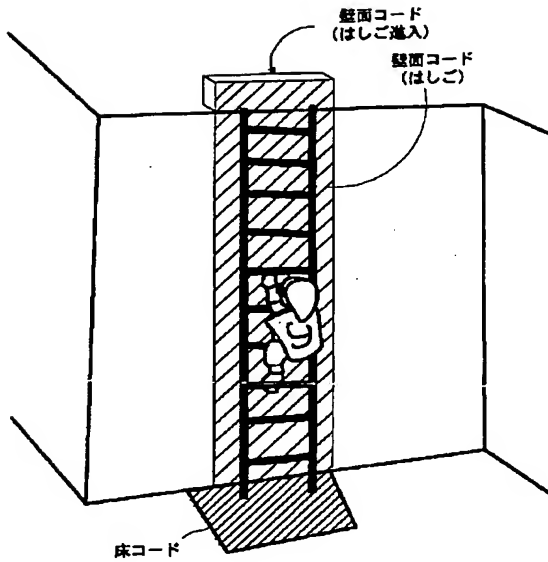
【図32】



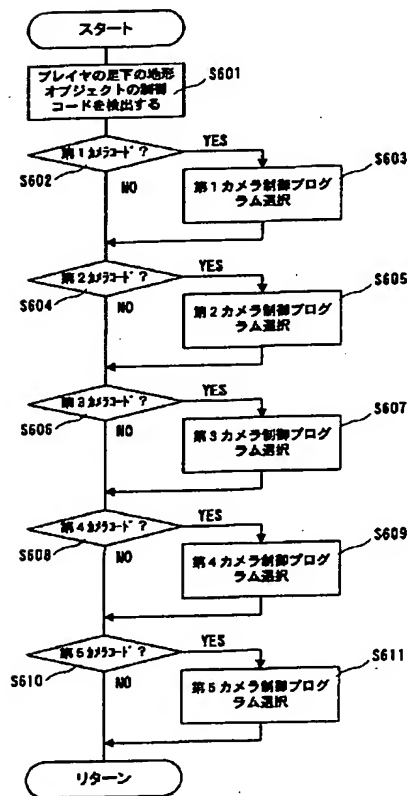
【図29】



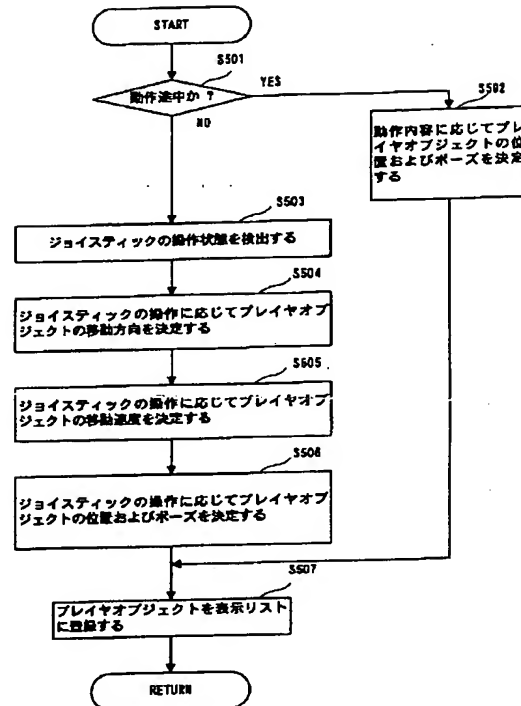
【図25】



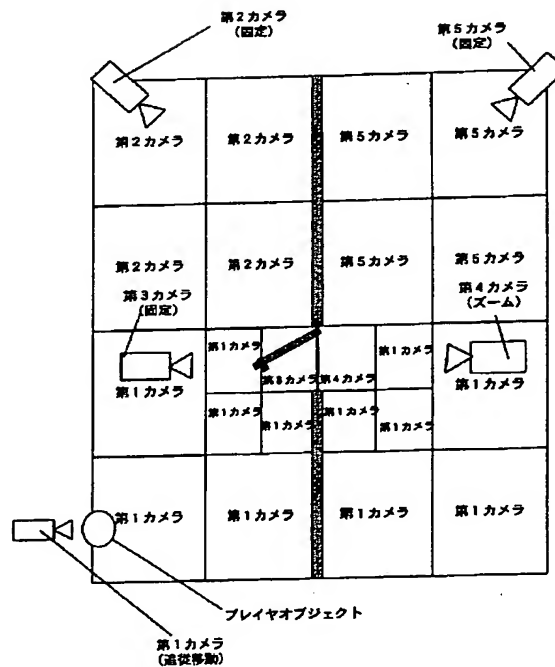
【図27】



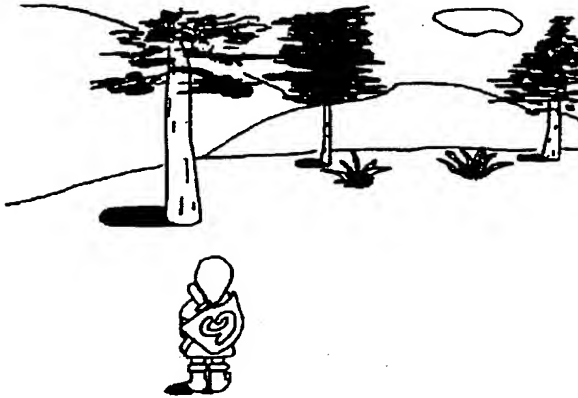
【図26】



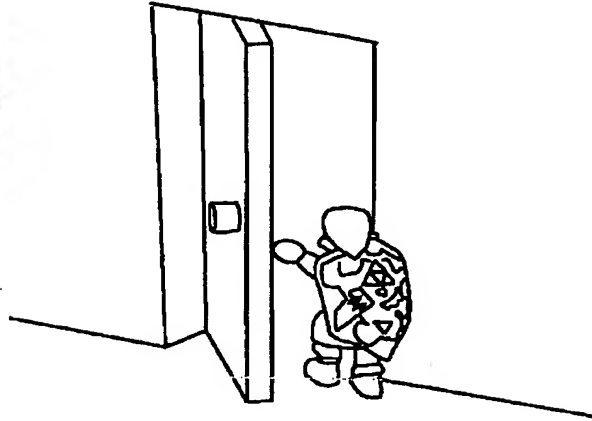
【図28】



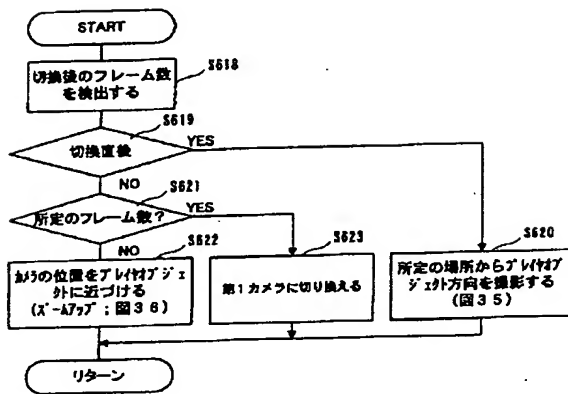
【図30】



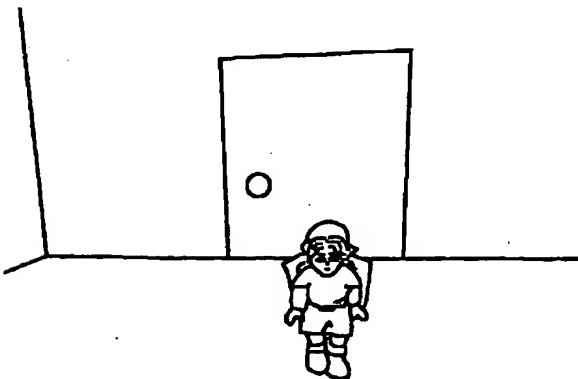
【図33】



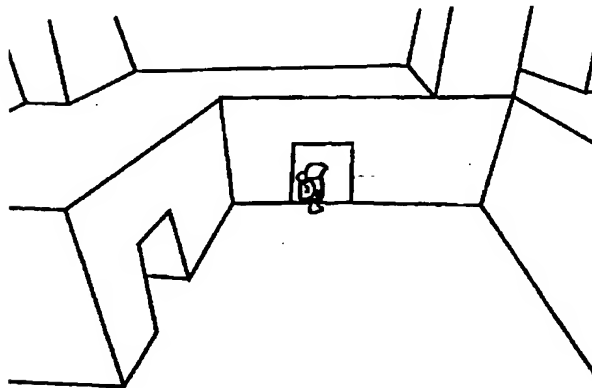
【図34】



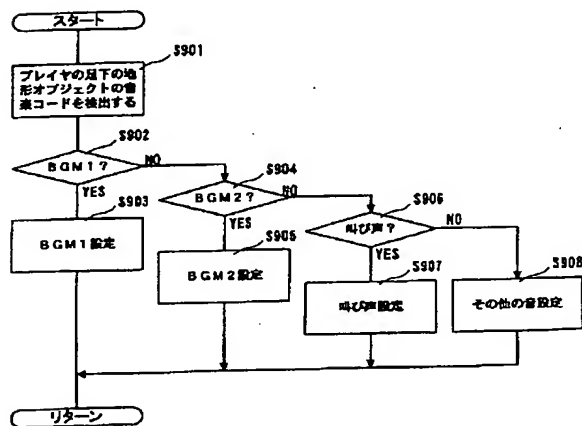
【図36】



【図35】



【図37】



フロントページの続き

(72)発明者 山田 洋一
京都府京都市東山区福稲上高松町60番地
任天堂株式会社内
(72)発明者 岩脇 敏夫
京都府京都市東山区福稲上高松町60番地
株式会社エス・アール・ディー内

Fターム(参考) 2C001 AA00 AA16 AA17 BA00 BA02
BA05 BC00 BC10 CA01 CA06
CB01 CB05 CC02 CC08
5B050 AA10 BA08 BA09 CA07 EA24
EA28 FA02 FA10